


Local device and process diagnostics in a process control network having distributed control functions

Patent number: DE69717838T
Publication date: 2003-09-25
Inventor: LARSON H (US); BURNS A (US); BROWN K (US)
Applicant: FISHER CONTROLS INTERNAT INC (US)
Classification:
- **international:** G05B19/042; G05B19/418
- **european:** G05B19/042N1; G05B19/418N
Application number: DE19976017838T 19971001
Priority number(s): US19960726262 19961004; US19970922938 19970903;
WO1997US17739 19971001

Also published as:

 WO9814848 (A1)
 EP0929850 (A1)
 US5970430 (A1)
 BR9712261 (A)
 EP0929850 (B1)

Report a data error here

Abstract not available for DE69717838T

Abstract of corresponding document: **US5970430**

A field device for use in a process control network having a plurality of devices communicatively coupled by a two-wire, all-digital communication bus includes a connector that connects the field device to the two-wire, all-digital bus to enable all-digital communication over the bus, a memory that stores a diagnostic test routine having a series of device or process diagnostic test instructions, and a controller that performs the device or process diagnostic test instructions stored in the memory to implement a device or process diagnostic test using the field device. A data collection unit within the field device collects diagnostic data generated during the diagnostic test and a communication unit communicates the collected diagnostic data over the bus to a host device for processing. The controller may include a program language interpreter adapted to interpret the diagnostic test instructions which may be provided to the field device from another one of the devices via the bus.

Data supplied from the **esp@cenet** database - Worldwide

THIS PAGE BLANK (USPTO)

5

01P21594



①9 BUNDESREPUBLIK
DEUTSCHLAND



DEUTSCHES
PATENT- UND
MARKENAMT

⑫ Übersetzung der
europäischen Patentschrift

⑨7 EP 0 929 850 B 1

⑩ DE 697 17 838 T 2

⑨1 Int. Cl.⁷:
G 05 B 19/042
G 05 B 19/418

②

②1 Deutsches Aktenzeichen:	697 17 838.2
⑧6 PCT-Aktenzeichen:	PCT/US97/17739
⑨6 Europäisches Aktenzeichen:	97 944 600.2
⑧7 PCT-Veröffentlichungs-Nr.:	WO 98/014848
⑧8 PCT-Anmeldetag:	1. 10. 1997
⑧7 Veröffentlichungstag der PCT-Anmeldung:	9. 4. 1998
⑨7 Erstveröffentlichung durch das EPA:	21. 7. 1999
⑨7 Veröffentlichungstag der Patenterteilung beim EPA:	11. 12. 2002
④7 Veröffentlichungstag im Patentblatt:	25. 9. 2003

⑩0 Unionspriorität:

726262	04. 10. 1996	US
922938	03. 09. 1997	US

⑦3 Patentinhaber:

Fisher Controls International Inc., Clayton, Mo., US

⑦4 Vertreter:

Meissner, Bolte & Partner, 80538 München

⑥4 Benannte Vertragsstaaten:

DE, FI, FR, GB, SE

⑦2 Erfinder:

LARSON, H., Brent, Marshalltown, US; BURNS, A.,
Harry, Marshalltown, US; BROWN, K., Larry,
Marshalltown, US

⑤4 LOKAL PLAZIERBARE ANORDNUNG UND PROZESSDIAGNOSTIK IN EINEM
PROZESSSTEUERUNGSNETZWERK MIT VERTEILTEN STEUERFUNKTIONEN

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II 5 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

DE 697 17 838 T 2

DE 697 17 838 T 2

697 17 838.2-08

**LOKALE GERÄTE- UND PROZESSDIAGNOSE IN EINEM PROZESS-
STEUERUNGSNETZ MIT VERTEILTEN STEUERUNGSFUNKTIONEN**

5 Verwandte Anmeldung

Dies ist eine Teilfortführungsanmeldung der US-Patentanmeldung Nr. 08/726,262, angemeldet am 4. Oktober 1996.

10 Gebiet der Erfindung

Die vorliegende Erfindung betrifft allgemein Prozeßsteuerungsnetze und insbesondere ein Verfahren und eine Vorrichtung zur Durchführung von Diagnosen an lokalen Geräten und an dem Prozeß in einem Prozeßsteuerungsnetz, das verteilte Steuerungsfunktionen hat.

Beschreibung des Stands der Technik

20 Großprozesse wie chemische, Öl- und andere Fertigungs- und Raffinationsprozesse umfassen zahlreiche Feldgeräte, die an den verschiedensten Stellen angeordnet sind, um Parameter des Prozesses zu messen und zu regulieren und dadurch die Steuerung des Prozesses zu bewirken. Diese Feldgeräte können beispielsweise Sensoren wie etwa Temperatur-, Druck- und Durchflußmengensensoren sowie Steuerelemente wie etwa Ventile und Schalter sein.

25 Historisch benutzte die Prozeßsteuerungsindustrie manuelle Vorgänge wie etwa die manuelle Ablösung von Pegel- und Druckmessern, das Drehen von Ventilrädern usw., um die Messung und Steuerung von Feldgeräten in einem Prozeß durchzuführen. Mit dem Beginn des zwanzigsten Jahrhunderts begann die Prozeßsteuerungsindustrie mit
30 der Anwendung der örtlichen pneumatischen Steuerung, wobei lokale pneumatische Steuergeräte, Meßumformer und Ventilpositionierer an verschiedenen Stellen innerhalb einer Prozeßanlage angeordnet wurden, um die Steuerung bestimmter Anlagenörtlich-

keiten durchzuführen. Mit dem Aufkommen des mikroprozessorbasierten verteilten Steuerungssystems bzw. DCS in den siebziger Jahren wurde die verteilte elektronische Prozeßsteuerung in der Prozeßsteuerungsindustrie vorherrschend.

5 Es ist bekannt, daß ein verteiltes Steuerungssystem bzw. DCS einen Analog- oder einen Digitalrechner wie etwa eine programmierbare logische Steuereinheit aufweist, die mit zahlreichen elektronischen Überwachungs- und Steuereinrichtungen wie etwa elektroni-

10 schen Sensoren, Meßumformern, Strom-Druck-Wandlern, Ventilpositionierern usw., die in einem Prozeß verteilt angeordnet sind, verbunden ist. Der DCS-Rechner speichert und implementiert ein zentrales und häufig komplexes Steuerungsschema, um die Mes-

15 sung und Steuerung von Einrichtungen innerhalb des Prozesses zu bewirken und dadurch Prozeßparameter entsprechend einem übergeordneten Steuerungsplan zu steuern. Gewöhnlich ist jedoch das von einem DCS implementierte Steuerungsschema Eigentum des Herstellers der DCS-Steuereinheit, was eine Erweiterung, ein Aufrüsten, eine Umprogrammierung und die Wartung des DCS schwierig und teuer macht, weil der

20 DCS-Provider zur Durchführung einer oder aller dieser Aktivitäten auf integrale Weise involviert werden muß. Ferner kann es sein, daß die Einrichtungen, die von einem bestimmten DCS verwendet oder in einem solchen angeschlossen werden können, aufgrund der firmeneigenen Beschaffenheit der DCS-Steuereinheit und durch die Tatsache beschränkt sein können, daß der Provider einer DCS-Steuereinheit eventuell bestimmte Einrichtungen oder Funktionen von Einrichtungen, die von anderen Herstellern kommen, nicht unterstützt.

25 Zur Überwindung einiger der Probleme, die mit der Anwendung von herstellereigenen DCS einhergehen, hat die Prozeßsteuerungsindustrie eine Reihe von offenen Standard-Kommunikationsprotokollen entwickelt wie beispielsweise die Protokolle HART®, PROFIBUS®, WORLDFIP®, LONWORKS®, Device-Net® und CAN, die die gemeinsame Verwendung von Feldgeräten, die von verschiedenen Herstellern stammen, innerhalb des-

30 selben Prozeßsteuerungsnetzes gestatten. Tatsächlich kann jedes Feldgerät, das einem dieser Protokolle entspricht, innerhalb eines Prozesses eingesetzt werden, um mit einer DCS-Steuereinheit oder einer anderen Steuereinheit, die das Protokoll unterstützt, zu kommunizieren und davon gesteuert zu werden, auch wenn dieses Feldgerät bzw. diese Feldeinrichtung von einem anderen als dem Hersteller der DCS-Steuereinheit stammt.

Außerdem gibt es heute eine Bewegung innerhalb der Prozeßsteuerungsindustrie in Richtung der Dezentralisierung der Prozeßsteuerung, wodurch DCS-Steuereinheiten vereinfacht werden oder die Notwendigkeit für DCS-Steuerungen weitgehend entfällt. Eine dezentrale Steuerung wird dadurch erhalten, daß man Prozeßsteuereinrichtungen wie etwa Ventilpositionierer, Meßumformer usw. eine oder mehrere Prozeßsteuerfunktionen ausführen läßt und dann Daten über eine Busstruktur zur Nutzung durch andere Prozeßsteuereinrichtungen bei der Durchführung anderer Steuerfunktionen überträgt. Zur Implementierung dieser Steuerfunktionen umfaßt jede Prozeßsteuereinrichtung einen Mikroprozessor, der imstande ist, eine oder mehrere Steuerfunktionen auszuführen sowie mit anderen Prozeßsteuereinrichtungen zu kommunizieren, wobei ein offenes Standard-Kommunikationsprotokoll angewandt wird. Auf diese Weise können von verschiedenen Herstellern stammende Feldgeräte innerhalb eines Prozeßsteuerungsnetzes miteinander verbunden werden, um miteinander zu kommunizieren und eine oder mehrere Prozeßsteuerfunktionen unter Bildung einer Steuerschleife ohne Eingriff durch eine DCS-Steuereinheit auszuführen. Das vollkommen digitale Zweidrahtbus-Protokoll, das heute von der Fieldbus Foundation verbreitet wird und als das FOUNDATION® Fieldbus-Protokoll (nachstehend "Fieldbus") bekannt ist, ist ein offenes Kommunikationsprotokoll, das es Einrichtungen, die von verschiedenen Herstellern stammen, erlaubt, über einen Standardbus miteinander betrieben zu werden und miteinander zu kommunizieren, um eine dezentrale bzw. verteilte Steuerung innerhalb eines Prozesses auszuführen.

Wie oben gesagt wird, vereinfacht die Dezentralisierung von Prozeßsteuerfunktionen eine DCS-Steuereinheit und eliminiert in manchen Fällen auch die Notwendigkeit für eine firmeneigene DCS-Steuereinheit, wodurch für einen Prozeßbetreiber wiederum die Notwendigkeit geringer wird, sich in bezug auf die Änderung oder Aufrüstung eines von der DCS-Steuereinheit implementierten Steuerschemas auf den Hersteller der DCS-Steuereinheit zu verlassen. Eine dezentrale Steuerung erschwert jedoch auch die Durchführung von Diagnosen wie etwa von Prozeßdiagnosen, die typischerweise von einer DCS-Steuereinheit durchgeführt werden. Die Durchführung von regelmäßigen Diagnosen wie etwa Geräte- und Prozeßdiagnosen ist aber sehr wichtig, wenn Feldgeräte wie etwa Fluidsteuerventile in rauen Umgebungen verwendet werden, in denen beispielsweise die Temperatur- und Druckbereiche innerhalb weiter Grenzen veränderlich sind. In solchen Umgebungen ist eine beträchtliche Wartung erforderlich, was die peri-

odische präventive Wartung, die Wartung aufgrund eines Ventilversagens sowie die Prüfung dahingehend, daß alle Ventile richtig funktionieren, umfaßt.

In einer Standard-DCS-Umgebung ist ein Computer (wie etwa ein PC) mit dem Netz verbunden und führt die Gerätediagnosen beispielsweise an einem Ventil oder einer Positionierer-/Ventil-Kombination aus, indem er ein Diagnosesteuersignal an den Positionierer sendet, der dann das Ventil durch einen Testhub oder Testzyklus treibt, der dem Diagnosesteuersignal zugeordnet ist. Während dieses Zeitraums mißt der Computer Ausgangswerte des Positionierers und/oder des Ventils wie etwa Änderungen der Ventilstellung, die aufgrund des Diagnosesteuersignals auftreten, und führt anschließend eine Analyse der ausgegebenen Meßwerte durch, um den Betriebszustand des Ventils oder der Positionierer-/Ventil-Einrichtung zu bestimmen.

Bei einem bekannten Diagnosesystem für Fluidsteuerventile wie etwa pneumatisch betätigte Ventile ist ein Drucksensor vorgesehen, um einen veränderlichen Druck am Eingang eines Ventils zu erfassen, und ein Positionssensor ist vorgesehen, um die Bewegung eines Ventilstopfens zu erfassen. Das Ventil wird durch einen Testbetriebszyklus angetrieben, indem dem Eingang eines pneumatischen Ventils ein gesteuerter veränderlicher pneumatischer Druck zugeführt wird. Beispielsweise während des Testbetriebszyklus einer dynamischen Abtastung wird der Ventilstopfen durch einen gewünschten Bereich bewegt, normalerweise von einer vollständig offenen Position in eine vollständig geschlossene Position und zurück aus der vollständig geschlossenen Position in die vollständig offene Position. Alternativ können schrittweise Tests den Ventilstopfen in einer Serie von Einzelschritten bewegen, um bestimmte Ventilparameter zu prüfen.

Während des Testbetriebszyklus liefert der Drucksensor ein Ausgangssignal, das dem veränderlichen Druck am Ventileingang entspricht, und der Positionssensor liefert ein Eingangssignal, das der Bewegung des Ventilstopfens entspricht. Die jeweiligen Eingangs- oder Ausgangssignale des Luftdrucks am Betätiger und die Ventilstopfen- oder Ventilspindelposition werden dann verarbeitet, um Daten zu gewinnen, die die Druckänderung am Ventileingang als eine Funktion der Bewegung des Ventilstopfens während des Testbetriebszyklus repräsentieren. Die Ventilspindelbelastung wird durch Multiplikation des Luftdrucks mit der Wirkfläche der Betätigermembran abgeleitet.

Das Diagnosesystem empfängt die Diagnosebefehle und übermittelt die Diagnoseinformation von den Sensoren auf einer Übertragungsleitung zu einer externen Steuerkonsole oder einem Prozessor/Computer. Die externe Steuerkonsole oder der externe Prozessor/Computer fordert einen einzelnen Diagnosetest an und wartet auf das Ergebnis, während das Diagnosesystem den Test ausführt. Nach beendetem Test sendet das Diagnosesystem das Testergebnis an die externe Steuerkonsole. Für jedes Ventil werden viele verschiedene Diagnoseprüfungen ausgeführt, und ein Steuerungssystem umfaßt im allgemeinen viele Ventile, so daß die Diagnoseprüfzeit lang sein kann.

Ferner es in einer Standard-DCS-Umgebung auch bekannt, daß ein Computer wie etwa eine DCS-Steuereinheit die Prozeßdiagnose beispielsweise unter Verwendung eines Ventils oder einer Positionierer-/Ventil-Kombination ausführt, indem er ein Diagnosesteuersignal an den Positionierer sendet, der das Ventil dann durch eine Testfolge antreibt.

In einer Standard-DCS-Umgebung kann die Diagnose an Geräten oder an dem Prozeß durchgeführt werden, ohne daß das System in erheblichem Umfang neu verdrahtet oder neu konfiguriert werden muß, weil die DCS-Steuereinheit oder der externe Computer bereits so konfiguriert ist, daß er die Sollwerte (oder andere Eingangswerte) der verschiedenen Geräte innerhalb des Prozesses steuert und Ausgangswerte der Geräte und andere Prozeßparameter mißt, um eine zum Normalbetrieb des Prozesses gehörende Steuerungsstrategie zu implementieren. Daher ist die Durchführung der Diagnose in einer Standard-DCS-Umgebung eigentlich eine Angelegenheit der Verwendung der DCS-Steuereinheit oder eines anderen externen Computers auf eine geringfügig verschiedene Weise, um eine oder mehrere Geräte innerhalb des Prozesses zu steuern, und der Verwendung der DCS-Steuereinheit oder des anderen externen Computers zur Messung von Prozeß- und Geräteparametern. In Standard-DCS-Umgebungen können daher Diagnoseroutinen in einer zentralen DCS-Steuereinheit oder einem anderen zentralen externen Computer gespeichert und genutzt werden, um nahezu jede Geräte- oder Prozeßdiagnose auszuführen, ohne daß das Prozeßsteuerungsnetz erheblich rekonfiguriert werden muß. Wegen der zentralisierten Beschaffenheit dieser Diagnoseroutinen liefern sie aber leider nicht viele detaillierte Informationen über Feldgeräte.

Bei einem Prozeßsteuerungsnetz, das verteilte Steuerfunktionen hat, ist eine zentrale Systemsteuereinheit, soweit sie vorhanden ist, nicht dazu konfiguriert, alle Feldgeräte innerhalb eines Prozesses einzeln zu steuern, und ist auch nicht zum Empfang von Daten konfiguriert, die sämtliche entsprechenden Geräte- oder Prozeßparameter betreffen, die zur Durchführung der Geräte- und Prozeßdiagnose notwendig sind. Statt dessen werden verschiedene Prozeßsteuerschleifen der Steuerungsstrategie von einer Reihe von kommunikativ gekoppelten Einrichtungen implementiert, die an verteilten Stellen innerhalb des Prozeßsteuerungsnetzes angeordnet sind. Typischerweise sind diese Einrichtungen dazu konfiguriert, geplante periodische Kommunikationen zu nutzen, um Daten zu übermitteln, die zur Implementierung der bestimmten Steuerfunktionen notwendig sind, die einer Prozeßsteuerschleife zugeordnet sind, und um andere Daten (wie etwa Sollwertänderungen) unter Nutzung von aperiodischen oder asynchronen Kommunikationen zu übermitteln. Infolgedessen ist in einem Prozeßsteuerungsnetz, das verteilte Steuerfunktionen hat, die unter Anwendung von planmäßigen periodischen Kommunikationen implementiert sind, ein Zentralrechner nicht imstande, ein strikt determiniertes Steuersignal an eine Prozeßsteuereinrichtung zu senden, solange das System zur Implementierung der normalen Steuerungsstrategie konfiguriert ist, weil der Hauptrechner asynchrone Kommunikationen nutzen muß, um das Diagnosesteuersignal zu übermitteln, und daher keine Möglichkeit hat, den genauen Zeitpunkt zu bestimmen, zu dem das Diagnosesteuersignal (oder die verschiedenen Anteile davon) an der zu prüfenden Einrichtung ankommt. Tatsächlich hat bei der Anwendung von asynchronen Kommunikationen ein Hauptrechner keine Möglichkeit zu wissen, wann das Diagnosesteuersignal (oder irgendein bestimmter Teil davon) tatsächlich an dem Eingang des zu steuernden Geräts ankommt. Damit also ein Hauptrechner ein determiniertes Diagnosesteuersignal an eine Einrichtung in einem Prozeßsteuerungsnetz mit verteilten Steuerfunktionen übermitteln kann, muß die Steuerungskonfiguration des Netzes rekonfiguriert werden, was bedingt, daß der Prozeß abgeschaltet wird.

Ferner können zwar manche Prozeßsteuerungseinrichtungen wie etwa die Fieldvue- und die Flowscanner-Einrichtungen, die von Fisher Controls International Inc. hergestellt und verkauft werden, eine Selbstdiagnose ausführen, diese Einrichtungen sind jedoch auf den Gebrauch in Prozeßsteuerungssystemen beschränkt, die ein analoges oder kombiniertes analoges/digitales Kommunikationsprotokoll für die Kommunikationen zwischen verschiedenen Einrichtungen verwenden. Derzeit gibt es keine Prozeßsteue-

rungs-Feldeinrichtung, die imstande ist, eine Selbstdiagnose in einem vollständig digitalen System wie etwa einem Fieldbus-System oder in einem Kommunikationssystem, das verteilte Steuerfunktionen ausführt, durchzuführen.

5 Ferner sind Prozeßsteuerungseinrichtungen, die eine Selbstdiagnose durchführen können, typischerweise darauf beschränkt, nur die Diagnosevorgänge auszuführen, die vom Hersteller der Einrichtung festcodiert wurden, und können daher keine Diagnose-
10 routinen oder -tests ausführen, die von einem Rechner oder einem Anwender erzeugt werden (und die Routinen aufweisen können, die von einem anderen Gerätehersteller entwickelt wurden). Diese Situation hindert einen Anwender daran, an sämtlichen ver-
15 schiedenenartigen Geräten innerhalb einer Anlage den gleichen Test ablaufen zu lassen.

20 Außerdem sind Prozeßsteuerungsgeräte, die eine Selbstdiagnose durchführen, im allgemeinen unfähig, eine Prozeßdiagnose durchzuführen. Somit muß eine Haupteinrichtung eingerichtet werden, um die Prozeßdiagnose durchzuführen, und zwar auch in
25 einem System, das Feldgeräte hat, die eine gewisse Selbstdiagnose (d. h. Gerätediagnose) durchführen können. Wie oben gesagt wird, ist es jedoch schwierig, daß eine Haupteinrichtung in einem Prozeßsteuerungssystem mit verteilten Funktionen eine Pro-
30 zeßdiagnose ausführt, weil die Steuerungskonfiguration neu konfiguriert werden muß, damit der Hauptrechner eine Einrichtung synchron steuern kann. Ferner kann die Anwendung der verschiedenen Steuerschemata während der Prozeßdiagnose zu Ergebnissen führen, die fehlerhaft oder ungenau in bezug auf das Steuerungsschema sind, das im Normalbetrieb des Prozesses abläuft. Außerdem sind Feldgeräte mit Diagnosefähigkeiten nicht imstande, andere Feldgeräte ohne lokale Diagnosefähigkeiten zu diagnostizieren.

30 DE-A-195 10 466 zeigt ein System, das Steuerfunktionen hat, die über verschiedene Logikeinrichtungen verteilt sind, die mit einem zentralen Controller verbunden sind, der wiederum die Aktivitäten der verschiedenen Logikeinrichtungen koordiniert.

D2 "Advanced systems simplify control", Machine design, vol. 68, no. 12, 11. Juli 1996, S. 118, 120, und WO-A-96/12993 zeigen die Anwendung von Fieldbusprotokollen und Fieldbusgeräten.

Zusammenfassung der Erfindung

Die vorliegende Erfindung richtet sich auf ein Verfahren und eine Vorrichtung zur Durchführung von Geräte- und Prozeßdiagnosen an und von einer bestimmten Prozeßsteuerungseinrichtung innerhalb eines Prozeßsteuerungsnetzes und bevorzugt innerhalb eines Prozeßsteuerungsnetzes, das verteilte Steuerfunktionen hat. Gemäß der vorliegenden Erfindung wird eine Diagnoseprüfroutine (die eine Geräte- oder eine Prozeß-Diagnoseprüfroutine sein kann) in einer Prozeßsteuerungseinrichtung gespeichert und von dieser implementiert zur Durchführung von Diagnosen an dieser Prozeßsteuerungseinrichtung, ohne daß es erforderlich ist, den zu dem Prozeßsteuerungsnetz gehörigen Steuerungsplan zu rekonfigurieren. Somit kann die Diagnoseprüfroutine gemäß der vorliegenden Erfindung implementiert werden, während gleichzeitig ein Prozeß im wesentlichen unter der gleichen Steuerungsstrategie gesteuert wird, die im Normalablauf des Prozesses implementiert ist. Außerdem kann die Geräte- oder Prozeß-Diagnoseprüfroutine, die von einer Prozeßsteuerungseinrichtung gemäß der vorliegenden Erfindung implementiert wird, von einem Anwender an einer Haupteinrichtung erzeugt und an die Prozeßsteuerungseinrichtung übertragen werden, bevor die Diagnoseprüfroutine durchgeführt wird, was es der Prozeßsteuerungseinrichtung erlaubt, jede gewünschte Diagnoseprüfroutine einschließlich Routinen zu implementieren, die von anderen Geräteherstellern geliefert werden.

Gemäß einem Aspekt der vorliegenden Erfindung weist ein Feldgerät, das in einem Prozeßsteuerungssystem verwendet werden kann, das eine Vielzahl von Feldgeräten hat, die über einen digitalen gespeisten Zweidrahtbus miteinander verbunden sind, folgendes auf: ein pneumatisch betätigtes Fluidsteuerventil, einen Positionierer, der über eine pneumatische Druckleitung mit dem Fluidsteuerventil gekoppelt ist, um einen pneumatischen Druck zu erzeugen, der bewirkt, daß das Fluidsteuerventil in eine Position zwischen einer offenen und einer geschlossenen Position bewegt wird, und einen Positionssensor, der mit dem Positionierer und dem Fluidsteuerventil gekoppelt ist, um die Position des Fluidsteuerventils zu erfassen. Ein Drucksensor ist mit der pneumatischen Druckleitung gekoppelt, um den an das Fluidsteuerventil angelegten pneumatischen Druck zu erfassen, und ein Strom-Druck-Meßwandler ist mit dem Positionierer durch die pneumatische Druckleitung gekoppelt, um den pneumatischen Druck in der pneumatischen Druckleitung als eine Funktion eines elektrischen Signals zu steuern.

Eine elektronische Steuereinheit ist mit dem Strom-Druck-Meßwandler, dem Drucksensor und dem Positionssensor gekoppelt und weist Steuerlogik auf, die das elektrische Signal auf der Basis von Rückkopplungssignalen, die einen von dem Drucksensor erfaßten Druck und eine von dem Positionssensor erfaßte Position bezeichnen, und auf der Basis des Feldgerätesteuersignals bestimmt. Ferner ist eine digitale Schnittstelle mit dem digitalen gespeisten Zweidrahtbus und mit der elektronischen Steuereinheit gekoppelt und umfaßt eine Schaltung zur Zuführung von Energie, die von dem gespeisten Bus abgeleitet ist, zu dem Feldgerät sowie eine Zweirichtungs-Kommunikationsschaltung, die Signale einschließlich des Feldgerätesteuersignals von dem Bus empfängt und Signale, die einen Status des Feldgeräts bezeichnen, an den Bus übermittelt.

Gemäß einem anderen Aspekt der vorliegenden Erfindung weist ein Feldgerät zum Gebrauch in einem Prozeßsteuerungsnetz, das eine Vielzahl von Geräten hat, die kommunikativ über einen vollständig digitalen Zweidraht-Kommunikationsbus miteinander gekoppelt sind, folgendes auf: einen Verbinder, der an den vollkommen digitalen Zweidrahtbus anschließbar ist, um die vollkommen digitale Kommunikation über den Bus zu ermöglichen, einen Speicher, in dem eine Diagnoseprüfroutine gespeichert ist, die eine Serie von Diagnoseprüfbefehlen hat, und eine Steuereinheit, die die in dem Speicher gespeicherten Diagnoseprüfbefehle ausführt, um eine Diagnoseprüfung unter Verwendung des Feldgeräts zu implementieren. Das Feldgerät weist ferner auf: eine Datensammeleinheit, die Diagnosedaten sammelt, die während der Diagnoseprüfung erzeugt werden, und eine Kommunikationseinheit, die die gesammelten Diagnosedaten in einem vollständig digitalen Format über den Bus weiterleitet.

Bevorzugt weist die Steuereinheit einen Programmspracheninterpretierer auf, der so ausgebildet ist, daß er eine Programmsprache interpretiert, und die Diagnoseprüfbefehle sind in der Programmsprache gespeichert und werden der Steuereinheit des Feldgeräts von einem zweiten der Vielzahl von Geräten über den Bus zugeführt. Ebenso können die Diagnoseprüfbefehle, falls gewünscht, eine Geräte- und/oder eine Prozeßdiagnose ausführen. Wenn die Diagnoseprüfbefehle eine Prozeßdiagnose bezeichnen, ist die Datensammeleinheit so ausgebildet, daß sie während der Diagnoseprüfung Daten empfängt, die von anderen Geräten erzeugt werden.

Gemäß noch einem weiteren Aspekt der vorliegenden Erfindung weist ein Feldgerät zum Gebrauch in einem Prozeßsteuerungsnetz, das eine Vielzahl von Geräten hat, die über einen Bus kommunikativ gekoppelt sind, folgendes auf: einen Speicher, in dem eine Diagnoseprüfroutine gespeichert ist, die eine Serie von Diagnoseprüfbefehlen hat, eine Gerätesteuerereinheit, die die in dem Speicher gespeicherten Diagnoseprüfbefehle ausführt, um eine Diagnoseprüfung zu implementieren, eine Datensammeleinheit, die während der Diagnoseprüfung erzeugte Diagnosedaten sammelt, und eine Kommunikationseinheit, die die Diagnoseprüfbefehle von einem zweiten von der Vielzahl von Geräten über den Bus empfängt, die empfangenen Diagnoseprüfbefehle in dem Speicher speichert und die gesammelten Diagnosedaten über den Bus weiterleitet.

Gemäß noch einem anderen Aspekt der vorliegenden Erfindung weist ein Feldgerät zum Gebrauch bei der Durchführung einer Prozeßdiagnoseprüfung in einem Prozeßsteuerungsnetz, das eine Vielzahl von über einen Bus kommunikativ gekoppelten Geräten hat, folgendes auf: einen Speicher, in dem eine Prozeßdiagnoseprüfroutine gespeichert ist, die eine Serie von Diagnoseprüfbefehlen hat, die von dem Feldgerät zu implementieren sind, und eine Gerätesteuerereinheit, die die in dem Speicher gespeicherten Prozeßdiagnoseprüfbefehle ausführt, um eine Prozeßdiagnoseprüfung zu implementieren. Die Feldeinrichtung weist ferner eine Datensammeleinheit auf, die von dem Feldgerät während der Prozeßdiagnoseprüfung erzeugte Diagnosedaten sammelt und weitere Prozeßdiagnosedaten von einem zweiten von der Vielzahl von Geräten über den Bus empfängt. Eine Kommunikationseinheit innerhalb des Feldgeräts übermittelt die gesammelten Diagnosedaten und die weiteren Prozeßdiagnosedaten über den Bus, nachdem die Prozeßdiagnoseprüfung abgeschlossen ist.

Die vorliegende Erfindung ist durch die Ansprüche 1 bis 40 definiert und betrifft ein Feldgerät bzw. eine Feldeinrichtung zur Verwendung in einem Prozeßsteuerungsnetz, das eine Vielzahl von Geräten bzw. Einrichtungen hat, die durch einen vollkommen digitalen Kommunikationsbus kommunizierend gekoppelt sind. Das Feldgerät weist eine Verbindereinrichtung auf, die mit dem vollständig digitalen Bus verbunden ist, um die vollständig digitale Kommunikation über den Bus zu ermöglichen. In einem Speicher ist eine Diagnoseprüfroutine gespeichert, die eine Serie von Diagnoseprüfbefehlen hat. Eine Steuereinheit führt die Diagnoseprüfbefehle, die in dem Speicher gespeichert sind, aus, um eine Diagnoseprüfung unter Verwendung des Feldgeräts zu implementieren.

Eine Datensammeleinheit sammelt Diagnosedaten, die während der Diagnoseprüfung erzeugt werden. Eine Kommunikationseinheit übermittelt die gesammelten Diagnosedaten über den Bus in einem vollständig digitalen Format.

5 Kurzbeschreibung der Zeichnungen

- Fig. 1 ist ein schematisches Blockbild eines Prozeßsteuerungsnetzes, das das
Fieldbusprotokoll verwendet;
- Fig.2 ist ein schematisches Blockbild eines Fieldbusgeräts, in dem ein Satz von
drei Funktionsblöcken vorhanden ist;
- Fig. 3 ist ein schematisches Blockbild, das die Funktionsblöcke innerhalb einiger
der Geräte des Prozeßsteuerungsnetzes von Fig. 1 zeigt;
- Fig. 4 ist eine schematische Darstellung einer Steuerschleife für eine typische
Prozeßsteuerschleife innerhalb des Prozeßsteuerungsnetzes von Fig. 1;
- Fig. 5 ist ein Zeitdiagramm für einen Makrozyklus eines Segments des Busses
des Prozeßsteuerungsnetzes von Fig. 1;
- Fig. 6 ist ein schematisches Blockbild, das ein digitales Feldgerät zeigt, das ei-
nen schleifengespeisten Zweidraht-Zweirichtungs-Positionierer zur digita-
len Kommunikation hat;
- Fig. 7 ist ein Blockbild, das eine geeignete Feldgerätesteuereinheit zur Verwen-
dung bei der Steuerung des digitalen Feldgeräts von Fig. 6 zeigt;
- Fig. 8 ist ein Flußdiagramm, das eine Vorgehensweise zur Durchführung von Di-
agnoseprüfungen zeigt;
- Fig. 9 ist ein Flußdiagramm, das ein Diagnoseprüfprotokoll zum Prüfen des digi-
talen Feldgeräts von Fig. 6 zeigt;

Fig. 10A,

10B und 10C sind Diagramme, die verschiedene Diagnoseprüfsignale zeigen, die zur Durchführung von Gerätediagnosen gemäß der vorliegenden Erfindung angewandt werden;

Fig. 11A

und 11B sind schematische Steuerschleifen, die einen Diagnosedatensammelfunktionsblock gemäß der vorliegenden Erfindung aufweisen; und

Fig. 12 ist ein Flußdiagramm, das ein Diagnoseprüfprotokoll zur Durchführung einer Prozeßdiagnoseprüfung unter Anwendung des Diagnosedatensammelfunktionsblocks von Fig. 11 zeigt.

Beschreibung der bevorzugten Ausführungsformen

Die Geräte- und Prozeßdiagnosen der vorliegenden Erfindung werden zwar im einzelnen in Verbindung mit einem Prozeßsteuerungsnetz beschrieben, das Prozeßsteuerfunktionen auf dezentrale oder verteilte Weise unter Verwendung einer Gruppe von Fieldbuseinrichtungen implementiert; es ist aber zu beachten, daß die Diagnosen der vorliegenden Erfindung mit Prozeßsteuerungsnetzen verwendbar sind, die verteilte Steuerfunktionen unter Verwendung anderer Arten von Feldgeräten und Kommunikationsprotokollen durchführen, was Protokolle, die mit von Zweidrahtbussen verschiedenen Bussen arbeiten, und Protokolle einschließt, die nur analoge oder sowohl analoge als auch digitale Kommunikationen unterstützen. Beispielsweise kann die Geräte- oder Prozeßdiagnose der vorliegenden Erfindung in jedem Prozeßsteuerungsnetz verwendet werden, das verteilte Steuerfunktionen ausführt, auch wenn dieses Prozeßsteuerungsnetz die HART-, PROFIBUS-Kommunikationsprotokolle usw. oder irgendwelche anderen Kommunikationsprotokolle verwendet, die es heute gibt oder die in Zukunft entwickelt werden. Ferner kann die Diagnose gemäß der vorliegenden Erfindung auch mit Standard-Prozeßsteuerungsnetzen verwendet werden, die keine verteilten Steuerfunktionen ausführen, wie etwa HART-Netzen usw., und kann innerhalb jeder gewünschten Prozeßsteuerungseinrichtung einschließlich Ventilen, Positionierern, Meßwertumformern usw. verwendet werden.

Bevor die Einzelheiten der Diagnoseeinrichtung der vorliegenden Erfindung erörtert werden, folgt eine allgemeine Beschreibung des Fieldbusprotokolls, von Feldgeräten, die gemäß diesem Protokoll konfiguriert sind, und der Art und Weise, wie die Kommunikation in einem Prozeßsteuerungsnetz stattfindet, das das Fieldbusprotokoll verwendet. Es versteht sich jedoch, daß das Fieldbusprotokoll zwar ein relativ neues volldigitales Kommunikationsprotokoll ist, das zur Verwendung in Prozeßsteuerungsnetzen entwickelt wurde, dieses Protokoll aber im Stand der Technik bekannt und in zahlreichen Artikeln, Broschüren und Gebrauchsanleitungen beschrieben ist, die u. a. von der Fieldbus Foundation, einer gemeinnützigen Organisation mit dem Sitz in Austin, Texas, veröffentlicht und verteilt werden. Insbesondere ist das Fieldbusprotokoll und die Art und Weise der Kommunikation mit das Fieldbusprotokoll verwendenden Geräten und die Speicherung von Daten darin im einzelnen in den Handbüchern der Fieldbus Foundation als Communications Technical Specification und User Layer Technical Specification beschrieben, die hier summarisch eingeführt werden.

Das Fieldbusprotokoll ist ein volldigitales, serielles Zweirichtungs-Kommunikationsprotokoll, das eine standardisierte physische Schnittstelle zu einer Zweidrahtschleife oder einem Zweidrahtbus bildet, der "Feld"-Geräte wie Sensoren, Betätiger, Steuereinheiten, Ventile usw. miteinander verbindet, die in einer Instrumenten- oder Prozeßsteuerungsumgebung etwa einer Fabrik oder einer Anlage vorhanden sind. Das Fieldbusprotokoll bildet tatsächlich ein LAN für Feldinstrumente (Feldgeräte) innerhalb einer Prozeßanlage, wodurch es diesen Feldgeräten ermöglicht wird, Steuerfunktionen an in einem Prozeß verteilten Stellen auszuführen und vor und nach der Durchführung dieser Steuerfunktionen miteinander zu kommunizieren, um eine Gesamtsteuerungsstrategie zu implementieren. Da das Fieldbusprotokoll die Verteilung von Steuerfunktionen in einem ganzen Prozeßsteuerungsnetz erlaubt, verringert es die Komplexität des zentralen Prozeßsteuergeräts, das einem DCS typischerweise zugeordnet ist, oder beseitigt sogar vollständig die Notwendigkeit für ein solches zentrales Prozeßsteuergerät.

Gemäß Fig. 1 kann ein Prozeßsteuerungsnetz 10, das das Fieldbusprotokoll verwendet, einen Hauptrechner 12 aufweisen, der mit einer Reihe von anderen Einrichtungen wie etwa einem Programmlogikcontroller (PLC) 13, einer Reihe von Steuereinheiten 14, einer weiteren Haupteinrichtung 15 und einer Gruppe von Feldgeräten 16, 18, 20, 22, 24, 26, 28, 30 und 32 über eine Zweidraht-Fieldbusschleife oder einen Zweidraht-

Fieldbus-Bus 34 verbunden ist. Der Bus 34 umfaßt verschiedene Abschnitte oder Segmente 34a, 34b und 34c, die durch Brückeneinrichtungen 30 und 32 voneinander getrennt sind. Jeder Abschnitt 34a, 34b und 34c verbindet eine Untergruppe der an dem Bus 34 angebrachten Geräte miteinander, um die Kommunikation zwischen den Geräten auf eine noch zu beschreibende Weise zu ermöglichen. Selbstverständlich ist das Netz von Fig. 1 nur beispielhaft, und es gibt viele andere Möglichkeiten, wie ein Prozeßsteuerungsnetz unter Anwendung des Fieldbusprotokolls konfiguriert sein kann. Typischerweise ist ein Konfigurierer in einer der Einrichtungen wie etwa dem Hauptrechner 12 angeordnet und dafür zuständig, jedes der Geräte einzurichten oder zu konfigurieren (wobei diese Geräte insofern "intelligente" Geräte sind, als jedes einen Mikroprozessor aufweist, der Kommunikations- und in einigen Fällen Steuerfunktionen ausführen kann) sowie zu erkennen, wenn neue Feldgeräte an den Bus 34 angeschlossen werden, wenn Feldgeräte von dem Bus 34 entfernt werden, einige der von den Feldgeräten 16 bis 32 erzeugten Daten zu empfangen und mit einem oder mehreren Anwenderterminals in Dialog zu treten, wobei diese in dem Hauptrechner 12 oder in jedem anderen Gerät sein können, das auf irgendeine Weise mit dem Hauptrechner 12 verbunden ist.

Der Bus 34 unterstützt oder erlaubt die rein digitale Zweirichtungs-Kommunikation und kann auch ein Energiesignal an eines oder alle damit verbundenen Geräte wie etwa die Feldgeräte 16 bis 32 liefern. Alternativ kann jedes oder können alle Feldgeräte 12 bis 32 ihre eigenen Energieversorgungen haben oder über gesonderte Leiter (nicht gezeigt) mit externen Energieversorgungen verbunden sein. Die Geräte 12 bis 32 sind in Fig. 1 mit dem Bus 34 in einer Standard-Busverbindung verbunden, wobei eine Vielzahl von Geräten mit dem gleichen Drahtpaar verbunden ist, das die Bussegmente 34a, 34b und 34c bildet; das Fieldbusprotokoll erlaubt jedoch andere Geräte-/Draht-Topologien einschließlich Punktverbindungen, wobei jedes Gerät mit einer Steuereinheit oder einem Hauptrechner über ein separates Zweidrahtpaar (ähnlich typischen 4-20-mA-DCS-Analogsystemen) und Baum- oder "Stich"-Verbindungen verbunden ist, wobei jedes Gerät mit einem gemeinsamen Punkt in einem Zweidrahtbus verbunden ist, der beispielsweise ein Verteilerkasten oder ein Abschlußbereich in einem der Feldgeräte innerhalb eines Prozeßsteuerungsnetzes sein kann.

Daten können über die verschiedenen Bussegmente 34a, 34b und 34c mit gleichen oder verschiedenen Kommunikations-Baudraten oder Geschwindigkeiten entsprechend dem Fieldbusprotokoll gesendet werden. Beispielsweise ermöglicht das Fieldbusprotokoll eine Kommunikationsgeschwindigkeit (H1) von 31,25 kbit/s, die beispielhaft von den Bussegmenten 34b und 34c von Fig. 1 verwendet wird, und eine Kommunikationsgeschwindigkeit von 1,0 Mbit/s und/oder 2,5 Mbit/s (H2), die typischerweise für die hochentwickelte Prozeßsteuerung, Ferneingabe/-ausgabe und Hochgeschwindigkeits-Fabrikautomatisierungsanwendungen verwendet wird und hier beispielhaft von dem Bussegment 34a von Fig. 1 verwendet wird. Ebenso können Daten über die Bussegmente 34a, 34b und 34c gemäß dem Fieldbusprotokoll unter Anwendung der Spannungsmodus-Zeichengabe oder der Strommodus-Zeichengabe gesendet werden. Selbstverständlich ist die maximale Länge jedes Segments des Busses 34 nicht streng begrenzt, sondern ist statt dessen durch die Kommunikationsgeschwindigkeit, den Kabeltyp, den Drahtdurchmesser, die Busleistungsoption usw. dieses Abschnitts bestimmt.

Das Fieldbusprotokoll klassifiziert die Geräte, die an den Bus 34 angeschlossen werden können, in drei Kategorien, und zwar Basisgeräte, Verbindungshauptgeräte und Brückengeräte. Basisgeräte (wie etwa die Geräte 18, 20, 24 und 28 von Fig. 1) können kommunizieren, d. h. Übertragungssignale auf den oder von dem Bus 34 senden und empfangen, können jedoch nicht die Reihenfolge oder den Zeitpunkt der Kommunikation bestimmen, die auf dem Bus 34 erfolgt. Verbindungshauptgeräte (wie etwa die Geräte 16, 22 und 26 sowie der Hauptrechner 12 von Fig. 1) sind Geräte, die über den Bus 34 kommunizieren und imstande sind, den Fluß und die zeitliche Steuerung von Kommunikationssignalen auf dem Bus 34 zu steuern. Brückeneinrichtungen (wie etwa die Einrichtungen 30 und 32 von Fig. 1) sind Einrichtungen, die dazu konfiguriert sind, auf einzelnen Segmenten oder Zweigen eines Fieldbus-Busses zu kommunizieren und diese zu verbinden, um größere Prozeßsteuerungsnetze zu schaffen. Falls gewünscht, können Brückengeräte eine Umwandlung zwischen unterschiedlichen Datengeschwindigkeiten und/oder unterschiedlichen Datenzeichengabeformaten, die auf den verschiedenen Segmenten des Busses 34 verwendet werden, bewirken, können Signale verstärken, die zwischen den Segmenten des Busses 34 gesendet werden, können die zwischen den verschiedenen Segmenten des Busses 34 fließenden Signale filtern und können nur diejenigen Signale durchlassen, die dazu bestimmt sind, von einem Gerät an

einem der Bussegmente, mit dem die Brücke gekoppelt ist, empfangen zu werden, und/oder können andere Aktionen ausführen, die notwendig sind, um verschiedene Segmente des Busses 34 miteinander zu verbinden. Brückengeräte, die Bussegmente verbinden, die mit unterschiedlichen Geschwindigkeiten betrieben werden, müssen Verbindungsmasterfähigkeiten an der Niedriggeschwindigkeits-Segmentseite der Brücke haben. Die Hauptrechner 12 und 15, der PCL 13 und die Steuereinheiten 14 können jede Art von Fieldbusgerät sein, sind jedoch typischerweise Verbindungsmastergeräte.

Jedes der Geräte 12 bis 32 ist imstande, über den Bus 34 zu kommunizieren und, was wichtig ist, ist imstande, unabhängig eine oder mehrere Prozeßsteuerfunktionen unter Nutzung von Daten auszuführen, die von dem Gerät, von dem Prozeß oder von einem anderen Gerät über Kommunikationssignale auf dem Bus 34 erhalten werden. Fieldbusgeräte sind also imstande, Teile einer Gesamtsteuerungsstrategie direkt zu implementieren, die bisher von einer zentralen digitalen Steuereinheit eines DCS ausgeführt wurden. Zur Ausführung von Steuerfunktionen weist jedes Fieldbusgerät einen oder mehrere standardisierte "Blöcke" auf, die in einem Mikroprozessor innerhalb des Geräts implementiert sind. Insbesondere weist jedes Fieldbusgerät einen Ressourcenblock auf und kann Null oder mehr Funktionsblöcke und Null oder mehr Wandlerblöcke aufweisen. Diese Blöcke werden als Blockobjekte bezeichnet.

Ein Ressourcenblock speichert und kommuniziert gerätespezifische Daten, die einige der Charakteristiken eines Fieldbusgeräts betreffen, wie etwa einen Gerätetyp, eine Angabe der Geräteüberprüfung und Angaben, wo weitere gerätespezifische Informationen innerhalb eines Speichers des Geräts erhalten werden können. Verschiedene Gerätehersteller können zwar unterschiedliche Datenarten in dem Ressourcenblock eines Feldgeräts speichern, aber jede dem Fieldbusprotokoll entsprechende Feldeinrichtung umfaßt einen Ressourcenblock, in dem einige Daten gespeichert werden.

Ein Funktionsblock definiert und implementiert eine Eingabefunktion, eine Ausgabefunktion oder eine Steuerfunktion, die dem Feldgerät zugeordnet ist, und daher werden Funktionsblöcke im allgemeinen als Eingabe-, Ausgabe- und Steuerfunktionsblöcke bezeichnet. In Zukunft können jedoch auch andere Kategorien von Funktionsblöcken wie etwa Hybridfunktionsblöcke existieren oder entwickelt werden. Jeder Eingabe- oder Ausgabefunktionsblock erzeugt mindestens einen Prozeßsteuereingang (wie etwa eine

Prozeßvariable von einer Prozeßmeßeinrichtung) oder Prozeßsteuerausgang (wie etwa eine Ventilposition, die zu einer Betätigungseinrichtung übertragen wird), während jeder Steuerfunktionsblock einen Algorithmus nutzt (der firmeneigen sein kann), um einen oder mehrere Prozeßausgänge aus einem oder mehreren Prozeßeingängen und
5 Steuereingängen zu erzeugen. Beispiele von Standardfunktionsblöcken umfassen Analogeingangs- bzw. AI-, Analogausgangs- bzw. AO-, Bias- bzw. B-, Steuerselektor- bzw. CS-, Diskreteingangs- bzw. DI-, Diskretausgangs- bzw. DO-, Manuell-Lader- bzw. ML-, Proportional/Differential- bzw. PI-, Proportional/Integral/Differential- bzw. PID-, Verhältnis- bzw. RA und Signalselektor- bzw. SS-Funktionsblöcke. Andere Arten von Funktionsblöcken existieren jedoch, und neue Arten von Funktionsblöcken können definiert
10 oder geschaffen werden, um in der Fieldbusumgebung wirksam zu sein.

Ein Wandlerblock koppelt die Ein- und Ausgänge eines Funktionsblocks mit lokalen Hardwareeinrichtungen wie etwa Sensoren und Gerätebetätigern, um es Funktionsblöcken zu ermöglichen, die Ausgänge lokaler Sensoren zu lesen und lokale Einrichtungen
15 anzuweisen, ein oder mehr Funktionen wie das Bewegen eines Ventilelements auszuführen. Wandlerblöcke enthalten typischerweise Information, die notwendig ist, um Signale zu interpretieren, die von einer lokalen Einrichtung geliefert werden, und um lokale Hardwaregeräte richtig zu steuern, beispielsweise Information, die den Typ einer
20 lokalen Einrichtung bezeichnet, Kalibrierinformation, die einer lokalen Einrichtung zugeordnet ist, usw. Ein einzelner Wandlerblock ist typischerweise jedem Eingangs- oder Ausgangsfunktionsblock zugeordnet.

Die meisten Funktionsblöcke können Alarm- oder Ereignisanzeigen auf der Basis vorbestimmter Kriterien erzeugen und sind imstande, in verschiedenen Moden unterschiedlich zu arbeiten. Im allgemeinen können Funktionsblöcke arbeiten in: einem Automatikmodus, wobei beispielsweise der Algorithmus eines Funktionsblocks automatisch wirksam ist; einem Bedienermodus, in dem der Eingang oder Ausgang etwa eines
25 Funktionsblocks manuell gesteuert wird; einem Betriebsunfähigkeitsmodus, in dem der Block unwirksam ist; einem Kaskadenmodus, in dem die Operation des Blocks durch den Ausgang eines anderen Blocks beeinflusst (bestimmt) wird; und einem oder mehreren Fernmoden, in denen ein entfernter Computer den Modus des Blocks bestimmt. Es
30 gibt jedoch auch andere Betriebsarten im Fieldbusprotokoll.

Wichtig ist, daß jeder Block imstande ist, mit anderen Blöcken in den gleichen oder anderen Feldgeräten über den Fieldbus-Bus 34 unter Verwendung von Standardnachrichtenformaten zu kommunizieren, die von dem Fieldbusprotokoll definiert sind. Infolgedessen können Kombinationen von Funktionsblöcken (in dem gleichen Gerät oder in verschiedenen Geräten) miteinander kommunizieren, um eine oder mehrere dezentrale Steuerschleifen zu bilden. So kann beispielsweise ein PID-Funktionsblock in einem Feldgerät über den Bus 34 angeschlossen werden, um einen Ausgang eines AI-Funktionsblocks in einem zweiten Feldgerät zu empfangen; Daten an einen AO-Funktionsblock in einem dritten Feldgerät zu liefern und einen Ausgang des AO-Funktionsblocks als Rückführung zu empfangen, um eine Prozeßsteuerschleife separat und getrennt von jeglichem DCS-Steuergerät zu erzeugen. Auf diese Weise bewegen Kombinationen von Funktionsblöcken Steuerfunktionen aus einer zentralen DCS-Umgebung heraus, was es DCS-Multifunktionscontrollern erlaubt, Überwachungs- oder Koordinierungsfunktionen auszuführen oder überhaupt zu entfallen. Ferner bilden Funktionsblöcke eine graphische, blockorientierte Struktur zur einfachen Konfiguration eines Prozesses und ermöglichen die Verteilung von Funktionen zwischen Feldgeräten verschiedener Lieferanten, weil diese Blöcke ein konsistentes Kommunikationsprotokoll verwenden.

Zusätzlich dazu, daß jedes Feldgerät Blockobjekte enthält und implementiert, weist es auch ein oder mehr andere Objekte einschließlich Verknüpfungsobjekte, Trendobjekte, Warnobjekte und Betrachtungsobjekte auf. Verknüpfungsobjekte definieren die Verknüpfungen zwischen den Eingängen und Ausgängen von Blöcken (wie etwa Funktionsblöcken) sowohl innerhalb des Feldgeräts als auch über den Fieldbus-Bus 34.

Trendobjekte erlauben die lokale Trendbildung von Funktionsblockparametern für den Zugriff durch andere Geräte wie etwa den Hauptrechner 12 oder die Controller 14 von Fig. 1. Trendobjekte halten kurzzeitige historische Daten, die irgendeinen, beispielsweise einen Funktionsblockparameter betreffen, und liefern diese Daten auf asynchrone Weise an andere Geräte oder Funktionsblöcke über den Bus 34. Warnobjekte melden Alarme und Ereignisse auf dem Bus 34. Diese Alarme oder Ereignisse können jedes Ereignis betreffen, das innerhalb eines Geräts oder eines der Blöcke eines Geräts auftritt. Betrachtungsobjekte sind vordefinierte Gruppierungen von Blockparametern, die

in Mensch-Maschine-Standardschnittstellen genutzt werden und von Zeit zu Zeit anderen Geräten zur Betrachtung übermittelt werden können.

Fig. 2 zeigt ein Feldbus-Gerät, das beispielsweise eines der Feldgeräte 16 bis 28 von Fig. 1 ist und hier drei Ressourcenblöcke 48, drei Funktionsblöcke 50, 51 und 52 und zwei Wandlerblöcke 53 und 54 aufweist. Der eine Funktionsblock 50 (der ein Eingangsfunktionsblock sein kann) ist über den Wandlerblock 53 mit einem Sensor 55 verbunden, der beispielsweise ein Temperatursensor, ein Sollwert-Anzeigesensor usw. sein kann. Der zweite Funktionsblock 51 (der ein Ausgangsfunktionsblock sein kann) ist durch den Wandlerblock 54 mit einer Ausgangseinrichtung wie etwa einem Ventil 56 verbunden. Der dritte Funktionsblock 52 (der ein Steuerfunktionsblock sein kann) hat ein ihm zugeordnetes Trendobjekt zur Trendbestimmung des Eingangsparameters des Funktionsblocks 52.

Verknüpfungsobjekte 58 definieren die Blockparameter jedes zugeordneten Blocks, und Warnobjekte 59 liefern Alarme oder Ereignismeldungen für jeden der zugehörigen Blöcke. Betrachtungsobjekte 60 sind jedem der Funktionsblöcke 50, 51 und 52 zugeordnet und umfassen oder gruppieren Datenlisten für die Funktionsblöcke, denen sie zugeordnet sind. Diese Listen enthalten Informationen, die für jede einer Gruppe von verschiedenen definierten Ansichten erforderlich sind. Natürlich ist Fig. 2 nur beispielhaft, und andere Anzahlen und Arten von Blockobjekten, Verknüpfungsobjekten, Warnobjekten, Trendobjekten und Betrachtungsobjekten können in jedem Feldgerät vorgesehen sein.

Es wird nun auf Fig. 3 Bezug genommen; dieses Blockbild des Prozeßsteuerungsnetzes 10, das die Geräte 16, 18 und 24 als Positionierer-/Ventilgeräte und die Geräte 20, 22, 26 und 28 als Wandlergeräte zeigt, verdeutlicht auch die Funktionsblöcke, die dem Positionierer/Ventil 16, dem Meßwandler 20 und der Brücke 30 zugeordnet sind. Wie Fig. 3 zeigt, umfaßt das Positionierer/Ventilgerät 16 einen Ressourcen- bzw. RSC-Block 61, einen Wandler- bzw. XDR-Block 62 und eine Reihe von Funktionsblöcken, umfassend einen Analogausgabe- bzw. AO-Funktionsblock 63, zwei PID-Funktionsblöcke 64 und 65 und einen Signalwähl- bzw. SS-Funktionsblock 69. Der Wandler 20 umfaßt einen Ressourcenblock 61, zwei Wandlerblöcke 62 und zwei Analogeingabe- bzw. AI-Funktionsblöcke 66 und 67. Ferner umfaßt die Brücke 30 einen Ressourcenblock 61 und einen PID-Funktionsblock 68.

Es ist ersichtlich, daß die verschiedenen Funktionsblöcke von Fig. 3 gemeinsam in einer Reihe von Steuerschleifen zusammenwirken können (durch Kommunikation über den Bus 34), und die Steuerschleifen, in denen die Funktionsblöcke des Positionierers/Ventils 16, des Wandlers 20 und der Brücke 30 liegen, sind in Fig. 3 durch einen Schleifenerkennungsblock bezeichnet, der mit jedem dieser Funktionsblöcke verbunden ist. Wie Fig. 3 zeigt, sind also der AO-Funktionsblock 63 und der PID-Funktionsblock 64 des Positionierers/Ventils 16 und der AI-Funktionsblock 66 des Wandlers 20 mit einer Steuerschleife LOOP1 verbunden, während der SS-Funktionsblock 69 des Positionierers/Ventils 16, der AI-Funktionsblock 67 des Wandlers 20 und der PID-Funktionsblock 68 der Brücke 30 in einer Steuerschleife LOOP2 miteinander verbunden sind. Der PID-Funktionsblock 65 des Positionierers/Ventils 16 ist in eine Steuerschleife LOOP3 eingefügt.

Die miteinander verbundenen Funktionsblöcke, die die Steuerschleife LOOP1 in Fig. 3 bilden, sind im einzelnen in dem Schema dieser Steuerschleife entsprechend Fig. 4 dargestellt. Wie Fig. 4 zeigt, ist die Steuerschleife LOOP1 vollständig durch Nachrichtenverbindungen zwischen dem AO-Funktionsblock 63 und dem PID-Funktionsblock 64 des Positionierers/Ventils 16 und dem AI-Funktionsblock 66 des Wandlers 20 (Fig. 3) gebildet. Das Steuerschleifenbild von Fig. 4 zeigt die Kommunikationsverbindungen zwischen diesen Funktionsblöcken unter Verwendung von Leitungen, die die Prozeß- und Steuereingänge und -ausgänge dieser Funktionsblöcke anschließen. Der Ausgang des AI-Funktionsblocks 66, der ein Prozeßmeßsignal oder ein Prozeßparametersignal aufweisen kann, ist somit kommunikativ über das Bussegment 34b mit dem Eingang des PID-Funktionsblocks 64 gekoppelt, der einen Ausgang hat, der ein Steuersignal aufweist, das kommunikativ mit einem Eingang des AO-Funktionsblocks 63 gekoppelt ist. Ein Ausgang des AO-Funktionsblocks 63, der ein Rückführungssignal aufweist, das beispielsweise die Position des Ventils 16 bezeichnet, ist mit einem Steuereingang des PID-Funktionsblocks 64 verbunden. Der PID-Funktionsblock 64 verwendet dieses Rückführungssignal zusammen mit dem Prozeßmeßsignal von dem AI-Funktionsblock 66, um die richtige Steuerung des AO-Funktionsblocks 63 zu implementieren. Natürlich können die Verbindungen, die durch die Linien in dem Steuerschleifendiagramm von Fig. 4 bezeichnet sind, intern innerhalb eines Feldgeräts hergestellt werden, wenn wie im Fall des AO- und des PID-Funktionsblocks 63 und 64 die Funktionsblöcke innerhalb dessel-

ben Feldgeräts (z. B. des Positionierers/Ventils 16) liegen, oder diese Verbindungen können über den Zweidraht-Kommunikationsbus 34 unter Verwendung von synchronen Standard-Fieldbuskommunikationen implementiert werden. Selbstverständlich werden von anderen Funktionsblöcken, die in anderen Konfigurationen kommunikativ miteinander verbunden sind, andere Steuerschleifen implementiert.

Zur Implementierung und Ausführung der Kommunikations- und Steueraktivitäten verwendet das Fieldbusprotokoll drei allgemeine Technologiekategorien, die als physische Schicht, als Kommunikationsstapel und als Benutzerschicht gekennzeichnet sind. Die Benutzerschicht umfaßt die Steuerungs- und Konfigurationsfunktionen, die in Form von Blöcken (wie Funktionsblöcken) und Objekten innerhalb einer bestimmten Prozeßsteuereinrichtung oder eines bestimmten Feldgeräts vorgesehen sind. Die Benutzerschicht ist typischerweise auf firmeneigene Weise durch den Gerätehersteller ausgelegt, muß aber imstande sein, Nachrichten entsprechend dem Standardnachrichtenformat zu empfangen und zu senden, das durch das Fieldbusprotokoll definiert ist, und von einem Benutzer auf übliche Weise konfiguriert zu werden. Die physische Schicht und der Kommunikationsstapel sind notwendig, um die Kommunikation zwischen verschiedenen Blöcken verschiedener Feldgeräte auf standardisierte Weise unter Verwendung des Zweidrahtbusses 34 zu bewirken, und kann nach dem wohlbekannten OSI-Schichtkommunikationsmodell ausgebildet sein.

Die physische Schicht, die der OSI-Schicht 1 entspricht, ist in jedes Feldgerät und den Bus 34 eingebettet und wirksam, elektromagnetische Signale, die von dem Fieldbus-Übertragungsmedium (dem Zweidrahtbus 34) empfangen werden, in Nachrichten umzuwandeln, die von dem Kommunikationsstapel des Feldgeräts genutzt werden können. Die physische Schicht kann man sich als den Bus 34 und die elektromagnetischen Signale vorstellen, die auf dem Bus 34 an den Eingängen und Ausgängen der Feldgeräte vorhanden sind.

Der Kommunikationsstapel, der in jedem Fieldbusgerät vorhanden ist, umfaßt eine Sicherungsschicht, die der OSI-Schicht 2 entspricht, eine Fieldbus-Zugriffsunterschicht und eine Fieldbus-Nachrichtenbezeichnungsschicht, die der OSI-Schicht 6 entspricht. Es gibt keine entsprechende Struktur für die OSI-Schichten 3 bis 5 in dem Fieldbusprotokoll. Die Anwendungsmöglichkeiten eines Fieldbusgeräts weisen jedoch eine Schicht 7

auf, während eine Benutzerschicht eine Schicht 8 ist, die in dem OSI-Protokoll nicht definiert sind. Jede Schicht in dem Kommunikationsstapel ist für die Codierung oder Decodierung eines Teils der Nachricht oder des Signals zuständig, das auf dem Fieldbus-Bus 34 übertragen wird. Infolgedessen addiert oder entfernt jede Schicht des Kommunikationsstapels bestimmte Anteile des Fieldbus-Signals wie Anfangshinweise, Startbegrenzer und Endbegrenzer und decodiert in einigen Fällen die bereinigten Anteile des Fieldbussignals, um zu identifizieren, wohin der Rest des Signals oder der Nachricht gesendet werden soll, oder ob das Signal verworfen werden sollte, weil es beispielsweise eine Nachricht oder Daten für Funktionsblöcke enthält, die nicht in dem empfangenden Feldgerät liegen.

Die Datensicherungsschicht steuert die Übertragung von Nachrichten auf den Bus 34 und verwaltet den Zugriff auf den Bus 34 in Übereinstimmung mit einem deterministischen zentralen Busabwickler, der als verbindungsaktiver Abwickler bezeichnet und nachstehend im einzelnen beschrieben wird. Die Datensicherungsschicht entfernt eine Präambel von den Signalen auf dem Übertragungsmedium und kann die empfangene Präambel nutzen, um den internen Taktgeber des Feldgeräts mit dem ankommenden Fieldbussignal zu synchronisieren. Ebenso wandelt die Datensicherungsschicht Nachrichten an dem Kommunikationsstapel in physische Fieldbussignale um und codiert diese Signale mit Taktinformation, um ein "synchrones serielles" Signal zu erzeugen, das für die Übertragung auf dem Zweidrahtbus 34 eine richtige Präambel hat. Während des Decodiervorgangs erkennt die Datensicherungsschicht Spezialcodes innerhalb der Präambel wie etwa Startbegrenzer und Endbegrenzer, um den Beginn und das Ende einer bestimmten Fieldbusnachricht zu identifizieren, und kann eine Prüfsumme ausführen, um die Integrität des Signals oder der Nachricht, die von dem Bus 34 empfangen wurde, zu verifizieren. Ebenso überträgt die Datensicherungsschicht Fieldbussignale auf dem Bus 34 durch Hinzufügen von Start- und Endbegrenzern zu Nachrichten an dem Kommunikationsstapel und durch Plazieren dieser Signale auf dem Übertragungsmedium zum richtigen Zeitpunkt.

Die Fieldbus-Nachrichtenbezeichnungsschicht erlaubt der Benutzerschicht (d. h. den Funktionsblöcken, Objekten usw. eines Feldgeräts), über den Bus 34 unter Anwendung eines Standardsets von Nachrichtenformaten zu kommunizieren, und beschreibt die Kommunikationsdienste, Nachrichtenformate und Protokollverhalten, die erforderlich

sind, um Nachrichten zu bilden, die auf den Kommunikationsstapel zu bringen und an die Benutzerschicht zu senden sind. Da die Fieldbus-Nachrichtenbezeichnungsschicht standardisierte Kommunikationen für die Benutzerschicht liefert, sind spezifische Fieldbus-Nachrichtenbezeichnungs-Kommunikationsdienste für jede Art von Objekt, die oben beschrieben wurde, definiert. Beispielsweise umfaßt die Fieldbus-Nachrichtenbezeichnungsschicht Objektwörterbuchdienste, die einem Benutzer erlauben, ein Objektwörterbuch eines Geräts zu lesen. Das Objektwörterbuch enthält Objektbeschreibungen, die jedes der Objekte (wie etwa Blockobjekte) eines Geräts beschreiben oder identifizieren. Die Fieldbus-Nachrichtenbezeichnungsschicht stellt auch Kontextverwaltungsdienste bereit, die einem Benutzer erlauben, Kommunikationsbeziehungen zu lesen und zu ändern, die als virtuelle Kommunikationsbeziehungen (VCR) bekannt sind und noch beschrieben werden und die einem oder mehreren Objekten eines Geräts zugeordnet sind. Ferner stellt die Fieldbus-Nachrichtenbezeichnungsschicht variable Zugangsdienste, Ereignisdienste, Hochlade- und Herunterladedienste und Programmaufrufdienste bereit, die sämtlich im Fieldbusprotokoll wohlbekannt sind und daher hier nicht im einzelnen beschrieben werden. Die Fieldbus-Zugangsunterschicht tabelliert die Fieldbus-Nachrichtenbezeichnungsschicht in die Datensicherungsschicht.

Um die Operation dieser Schichten zu erlauben oder zu ermöglichen, weist jedes Fieldbusgerät eine Verwaltungs- bzw. Managementinformationsbasis bzw. MIB auf, die eine Datenbank ist, in der VCR, dynamische Variablen, Statistiken, Zeitpläne des verbindungsaktiven Abwicklers, Funktionsblock-Ausführungszeitpläne und Gerätekennungs- und -adreßinformationen gespeichert sind. Natürlich kann jederzeit unter Verwendung von Standard-Fieldbusnachrichten oder -befehlen Zugriff auf die Information innerhalb der MIB erfolgen. Außerdem wird gewöhnlich mit jedem Gerät eine Gerätebeschreibung geliefert, um einem Benutzer oder Hauptrechner eine erweiterte Ansicht der Information im VFD zu geben. In einer Gerätebeschreibung, die typischerweise markiert bzw. mit einem Token versehen sein muß, um von einem Hauptrechner genutzt zu werden, ist Information gespeichert, die der Hauptrechner benötigt, um die Bedeutung der Daten in den VFD eines Geräts zu verstehen.

Es versteht sich, daß zur Implementierung jeder Steuerungsstrategie unter Verwendung von Funktionsblöcken, die in einem Prozeßsteuerungsnetz verteilt sind, die Ausführung der Funktionsblöcke in bezug auf die Ausführung anderer Funktionsblöcke in

einer bestimmten Steuerschleife präzise geplant sein muß. Ebenso muß die Kommunikation zwischen verschiedenen Funktionsblöcken auf dem Bus 34 exakt geplant sein, so daß für jeden Funktionsblock vor dessen Ausführung die richtigen Daten geliefert werden.

5

Die Art und Weise, wie verschiedene Feldgeräte (und verschiedene Blöcke in Feldgeräten) über das Fieldbus-Übertragungsmedium miteinander kommunizieren, wird nachstehend unter Bezugnahme auf Fig. 1 beschrieben. Damit eine Kommunikation stattfindet, ist eines der Verbindungshauptgeräte an jedem Segment des Busses 34 (beispielsweise der Geräte 12, 16 und 26) als verbindungsaktiver Abwickler bzw. LAS (=link active scheduler) wirksam, der die Kommunikation auf dem zugehörigen Segment des Busses 34 aktiv abwickelt und steuert. Der LAS für jedes Segment des Busses 34 speichert und aktualisiert einen Kommunikationsplan (einen verbindungsaktiven Plan), der die Zeitpunkte enthält, zu denen jeder Funktionsblock jedes Geräte planmäßig mit dem Beginn der periodischen Kommunikationsaktivität auf dem Bus 34 beginnt, und die Zeitdauer enthält, während der diese Kommunikationsaktivität stattfinden soll. Es kann zwar an jedem Segment des Busses 34 ein, und zwar nur ein aktives LAS-Gerät geben, aber andere Verbindungshauptgeräte (wie etwa das Gerät 22 an dem Segment 34b) können als Backup-LAS dienen und aktiv werden, wenn beispielsweise der aktuelle LAS ausfällt. Basiseinrichtungen haben zu keiner Zeit die Fähigkeit, ein LAS zu werden.

10

15

20

25

30

Allgemein gesagt sind die Kommunikationsaktivitäten über den Bus 34 in sich wiederholende Makrozyklen unterteilt, von denen jeder eine synchrone Kommunikation für jeden Funktionsblock, der an einem bestimmten Segment des Busses 34 aktiv ist, und eine oder mehrere asynchrone Kommunikationen für einen oder mehrere der Funktionsblöcke oder Geräte, die an einem Segment des Busses 34 aktiv sind, aufweist. Ein Gerät kann aktiv sein, d. h. Daten zu jedem Segment des Busses 34 senden und Daten davon empfangen, auch wenn es physisch mit einem anderen Segment des Busses 34 verbunden ist, und zwar durch koordinierte Operation der Brücken und der LAS an dem Bus 34.

Während jedes Makrozyklus führt jeder der Funktionsblöcke, der an einem bestimmten Segment des Busses 34 aktiv ist, seine Funktion gewöhnlich zu einem jeweils anderen,

jedoch präzise geplanten (synchronen) Zeitpunkt aus und veröffentlicht zu einem anderen präzise geplanten Zeitpunkt seine Ausgangsdaten auf diesem Segment des Busses 34 als Reaktion auf einen Zwangsdatenbefehl, der von dem jeweiligen LAS erzeugt wird. Bevorzugt ist die Abwicklung jedes Funktionsblocks derart, daß er seine Ausgangsdaten kurz nach dem Ende der Ausführungsperiode des Funktionsblocks veröffentlicht. Ferner sind die Datenveröffentlichungszeitpunkte der verschiedenen Funktionsblöcke serienmäßig geplant, so daß nicht zwei Funktionsblöcke auf einem bestimmten Segment des Busses 34 gleichzeitig Daten veröffentlichen. Während der Zeitdauer, in der keine synchrone Kommunikation stattfindet, darf jedes Feldgerät nacheinander Warndaten, Ansichtsdaten usw. auf asynchrone Weise unter Anwendung von token-gesteuerten Kommunikationen übertragen. Die Ausführungszeiten und der Zeitaufwand, der erforderlich ist, um die Ausführung jedes Funktionsblocks zu vervollständigen, sind in der Managementinformationsbasis (MIB) des Geräts gespeichert, in dem sich der Funktionsblock befindet, während die Zeiten zum Senden der Zwangsdatenbefehle an jedes der Geräte auf einem Segment des Busses 34 in der MIB des LAS-Geräts für dieses Segment gespeichert sind, wie oben gesagt wird. Diese Zeiten sind typischerweise als versetzte Zeiten gespeichert, weil sie die Zeitpunkte kennzeichnen, zu denen ein Funktionsblock ausführen soll oder Daten als versetzte Daten vom Beginn eines "absoluten Abwicklungsplan-Startzeitpunkts" senden soll, der sämtlichen Geräten bekannt ist, die mit dem Bus 34 verbunden sind.

Um während jedes Makrozyklus Kommunikationen herzustellen, sendet der LAS, beispielsweise der LAS 16 des Bussegments 34b, einen Zwangsdatenbefehl an jedes der Geräte an dem Bussegment 34b entsprechend der Übertragungszeitenliste, die in dem verbindungsaktiven Plan gespeichert ist. Bei Empfang eines Zwangsdatenbefehls veröffentlicht ein Funktionsblock eines Geräts seine Ausgangsdaten auf dem Bus 34 für einen bestimmten Zeitraum. Da jeder der Funktionsblöcke typischerweise so eingeplant ist, daß seine Ausführung derart erfolgt, daß die Ausführung dieses Blocks beendet ist, kurz bevor der Block planmäßig einen Zwangsdatenbefehl empfangen soll, sollten die aufgrund eines Zwangsdatenbefehls veröffentlichten Daten die neuesten Ausgangsdaten des Funktionsblocks sein. Wenn jedoch ein Funktionsblock langsam ausführt und keine neuen Ausgänge im Zwischenspeicher hat, wenn er den Zwangsdatenbefehl empfängt, veröffentlicht der Funktionsblock die während des letzten Ablaufs des Funktions-

blocks erzeugten Daten und zeigt durch einen Zeitstempel an, daß die veröffentlichten Daten alte Daten sind.

Nachdem der LAS an jeden der Funktionsblöcke an einem bestimmten Segment des Busses 34 einen Zwangsdatenbefehl gesendet hat und während der Zeitdauern, in denen Funktionsblöcke ausführen, kann der LAS veranlassen, daß asynchrone Kommunikationsaktivitäten stattfinden. Zur Durchführung einer asynchronen Kommunikation sendet der LAS eine Tokenübergabenachricht an ein bestimmtes Feldgerät. Wenn ein Feldgerät eine Tokenübergabenachricht empfängt, hat dieses Feldgerät vollen Zugang zu dem Bus 34 (oder einem Segment davon) und kann asynchrone Nachrichten wie etwa Alarmnachrichten, Trenddaten, vom Bediener veranlaßte Sollwertänderungen usw. senden, bis die Nachrichten komplett sind oder eine maximale zugewiesene "Tokenhaltezeit" abgelaufen ist. Danach gibt das Feldgerät den Bus 34 (oder ein bestimmtes Segment davon) frei, und der LAS sendet eine Tokenübergabenachricht an ein anderes Gerät. Dieser Vorgang wiederholt sich bis zum Ende des Makrozyklus oder bis der LAS gemäß Plan einen Zwangsdatenbefehl aussendet, um die synchrone Kommunikation auszuführen. Selbstverständlich kann es in Abhängigkeit vom Umfang des Nachrichtenverkehrs und der Anzahl von Geräten und Blöcken, die mit einem bestimmten Segment des Busses 34 gekoppelt sind, geschehen, daß nicht jedes Gerät während eines Makrozyklus eine Tokenübergabenachricht empfangen kann.

Fig. 5 zeigt ein Zeitschema, das die Zeiten angibt, zu denen Funktionsblöcke an dem Bussegment 34b von Fig. 1 während jedes Makrozyklus des Bussegments 34b ausführen, und die Zeiten angibt, zu denen während jedes Makrozyklus, der dem Bussegment 34b zugeordnet ist, synchrone Kommunikationen stattfinden. In dem Zeitplan von Fig. 5 ist die Zeit auf der Horizontalachse eingetragen, und Aktivitäten, die zu den verschiedenen Funktionsblöcken des Positionierers/Ventils 16 und des Wandlers 20 (von Fig. 3) gehören, sind auf der Vertikalachse angegeben. Die Steuerschleife, in der jeder der Funktionsblöcke wirksam ist, ist in Fig. 5 tiefgestellt angegeben. So bezieht sich AI_{LOOP1} auf den AI-Funktionsblock 66 des Wandlers 20, PID_{LOOP1} bezieht sich auf den PID-Funktionsblock 64 des Positionierers/Ventils 16 usw. Die Blockausführungsperiode jedes der gezeigten Funktionsblöcke ist als schraffiertes Kästchen gezeigt, wogegen jede planmäßige synchrone Kommunikation als vertikaler Balken in Fig. 5 gekennzeichnet ist.

Gemäß dem Zeitplan von Fig. 5 erfolgt also während eines bestimmten Makrozyklus des Segments 34b (Fig. 1) zuerst die Ausführung des AI_{LOOP1} -Funktionsblocks während des durch das Kästchen 70 bezeichneten Zeitraums. Während der durch den vertikalen Balken 72 bezeichneten Zeit wird dann der Ausgang des AI_{LOOP1} -Funktionsblocks auf dem Bussegment 34b als Reaktion auf einen Zwangsdatenbefehl von dem LAS für das Bussegment 34b veröffentlicht. Ebenso bezeichnen die Kästchen 74, 76, 78, 80 und 81 die Ausführungszeiten der Funktionsblöcke PID_{LOOP1} , AI_{LOOP2} , AO_{LOOP1} , SS_{LOOP2} und PID_{LOOP3} (die für jeden der verschiedenen Blöcke verschieden sind), während die vertikalen Balken 82, 84, 86, 88 und 89 die Zeiten bezeichnen, zu denen die jeweiligen Funktionsblöcke PID_{LOOP1} , AI_{LOOP2} , AO_{LOOP1} , SS_{LOOP2} und PID_{LOOP3} Daten auf dem Bussegment 34b veröffentlichen.

Wie ersichtlich ist, zeigt der Zeitplan von Fig. 5 auch die Zeiten, die für asynchrone Kommunikationsaktivitäten verfügbar sind und während der Ausführungszeiten jedes der Funktionsblöcke und während der Zeit am Ende des Makrozyklus, in der keine Funktionsblöcke ausführen, und wenn keine synchrone Kommunikation auf dem Bussegment 34b erfolgt, stattfinden können. Falls gewünscht, können natürlich andere Funktionsblöcke gewollt so eingeplant sein, daß sie gleichzeitig ausführen, und es müssen nicht sämtliche Funktionsblöcke Daten auf dem Bus veröffentlichen, wenn beispielsweise kein anderes Gerät an den von einem Funktionsblock erzeugten Daten teilnimmt.

Feldgeräte sind imstande, Daten oder Nachrichten auf dem Bus 34 unter Verwendung von drei virtuellen Kommunikationsbeziehungen (VCR) zu veröffentlichen, die in der Fieldbus-Zugangsunterschicht des Stapels jedes Feldgeräts definiert sind. Ein Client-/Server-VCR dient für wartende, ungeplante, vom Benutzer initiierte Eins-zu-Eins-Kommunikationen zwischen Geräten an dem Bus 34. Solche zwischengespeicherten Nachrichten werden in der zur Übertragung eingegangenen Reihenfolge entsprechend ihrer Priorität ohne Überschreiben vorhergehender Nachrichten gesendet und empfangen. So kann ein Feldgerät eine Client/Server-VCR verwenden, wenn es eine Tokenübergabemessage von einem LAS empfängt, eine Anfragenachricht an ein anderes Gerät an dem Bus 34 zu senden. Der Anfragende wird als "Client" und das Gerät, das die Anfrage empfängt, als "Server" bezeichnet. Der Server sendet eine Antwort, wenn er ei-

ne Tokenübergabenachricht von dem LAS empfängt. Die Client/Server-VCR wird beispielsweise angewandt, um vom Bediener ausgelöste Anforderungen wie etwa Sollwertänderungen, Zugang zu und Änderungen von Abstimmparametern, Alarmbestätigungen und Up- und Downloads von Geräten auszuführen.

5

Eine Berichtsverteilungs-VCR wird für wartende, ungeplante, vom Benutzer ausgelöste Eins-an-viele-Kommunikationen verwendet. Wenn beispielsweise ein Feldgerät mit einem Ereignis- oder einem Trendbericht eine Tokenübergabenachricht von einem LAS empfängt, sendet dieses Feldgerät seine Nachricht an eine "Gruppenadresse", die in der Fieldbus-Zugangsunterschicht des Kommunikationsstapels dieses Geräts definiert ist. Geräte, die so konfiguriert sind, daß sie diese VCR mithören, empfangen den Bericht. Der Berichtsverteilungs-VCR-Typ wird charakteristisch von Fieldbusgeräten verwendet, um Alarmbenachrichtigungen an Bedienerkonsolen zu senden.

10

15

Ein Veröffentlichler/Teilnehmer-VCR-Typ dient für gepufferte Eins-an-viele-Kommunikationen. Gepufferte Kommunikationen sind solche, die nur die neueste Version der Daten speichern und senden, und daher werden vorhergehende Daten von neuen Daten vollständig überschrieben. Beispielsweise weisen Funktionsblockausgänge gepufferte Daten auf. Ein "Veröffentlichler"-Feldgerät veröffentlicht oder sendet eine Nachricht unter Verwendung des Veröffentlichler/Teilnehmer-VCR-Typs an sämtliche "Teilnehmer"-Feldgeräte an dem Bus 34, wenn das Veröffentlichler-Gerät eine Zwangsdatennachricht von dem LAS oder von einem Teilnehmergerät empfängt. Die Veröffentlichler/Teilnehmer-Beziehungen sind vorbestimmt und in der Fieldbus-Zugangsunterschicht des Kommunikationsstapels jedes Feldgeräts definiert und gespeichert.

20

25

Um die richtigen Kommunikationsaktivitäten über den Bus 34 sicherzustellen, sendet jeder LAS periodisch eine Zeitverteilungsnachricht an alle diejenigen Feldgeräte, die mit einem Segment des Busses 34 verbunden sind, was es den empfangenden Geräten erlaubt, ihre lokale Anwendungszeit miteinander zu synchronisieren. Zwischen diesen Synchronisierungsnachrichten wird die Taktzeit unabhängig in jedem Gerät auf der Basis seiner eigenen internen Uhr aufrechterhalten. Die Synchronisierung der Uhren erlaubt den Feldgeräten, Daten im gesamten Fieldbusnetz mit Zeitstempel zu versehen, um beispielsweise anzuzeigen, wann Daten erzeugt wurden.

30

Ferner ist in jedem LAS (und anderen Verbindungshauptgeräten) an jedem Bussegment eine "aktuelle Liste" gespeichert, die eine Liste sämtlicher Geräte ist, die mit diesem Segment des Busses 34 verbunden sind, d. h. sämtlicher Geräte, die auf eine Tokenübergabenachricht richtig antworten. Der LAS erkennt ständig neue Geräte, die einem Bussegment hinzugefügt werden, indem er periodisch Suchknotennachrichten an Adressen schickt, die nicht auf der aktiven Liste sind. Tatsächlich wird von jedem LAS verlangt, daß er mindestens eine Adresse absucht, nachdem er einen Zyklus des Sendens von Tokenübergabenachrichten an sämtliche Feldgeräte auf der aktiven Liste abgeschlossen hat. Wenn an der abgesuchten Adresse ein Feldgerät anwesend ist und die Suchknotennachricht empfängt, sendet das Gerät sofort eine Suchantwortnachricht zurück. Bei Empfang einer Suchantwortnachricht fügt der LAS das Gerät der aktiven Liste hinzu und bestätigt dies durch Senden einer Knotenaktivierungsnachricht an das abgesuchte Feldgerät. Ein Feldgerät bleibt so lange auf der aktiven Liste, wie dieses Feldgerät ordnungsgemäß auf Tokenübergabenachrichten anspricht. Ein LAS entfernt jedoch ein Feldgerät von der aktiven Liste, wenn das Feldgerät nach drei aufeinanderfolgenden Versuchen das Token nicht benutzt oder das Token nicht sofort an den LAS zurücksendet. Wenn der aktiven Liste ein Feldgerät hinzugefügt oder aus ihr entfernt wird, sendet der LAS Änderungen der aktiven Liste an sämtliche anderen Verbindungshauptgeräte an dem entsprechenden Segment des Busses 34, damit jedes Verbindungshauptgerät eine aktuelle Kopie der aktuellen Liste aufrechterhalten kann.

Wie oben erwähnt wurde, werden die Kommunikationsverbindungen zwischen den Feldgeräten und ihren Funktionsblöcken von einem Benutzer bestimmt und innerhalb des Prozeßsteuerungsnetzes 10 implementiert, indem eine Konfigurationsanwendung verwendet wird, die beispielsweise in dem Hauptrechner 12 vorhanden ist. Nach dem Konfigurieren arbeitet das Prozeßsteuerungsnetz 10 jedoch ohne Rücksicht auf Geräte- oder Prozeßdiagnosen und tritt somit mit dem Hauptrechner 12 in Dialog, um Standard-E/A-Funktionen, jedoch keine Diagnosefunktionen auszuführen.

Wenn ein Benutzer Diagnosen ausführen möchte, kann der Benutzer veranlassen, daß der Hauptrechner 12 Sollwertänderungen beispielsweise an den AO-Funktionsblock 63 der Steuerschleife LOOP1 sendet und die Rückmeldung in dem AO-Funktionsblock 63 unter Verwendung eines dem AO-Funktionsblock 63 zugeordneten Trendobjekts auf-

zeichnet. Zur Durchführung dieser Art von Kommunikation muß der Hauptrechner 12 jedoch asynchrone (nicht veröffentlichte) Kommunikationen verwenden, die dem Hauptrechner 12 Zugang zu dem Bus 34 nur dann gestatten, wenn der Hauptrechner 12 von einem LAS eine Tokenübergabenachricht empfängt. Infolgedessen kann es sein, daß die verschiedenen Teile des von dem Hauptrechner 12 erzeugten Diagnosesignals den AO-Funktionsblock 63 nicht zu präzise festgelegten (oder präzise geplanten) Zeiten erreichen, was bedeutet, daß das an dem AO-Funktionsblock 63 empfangene Diagnosesignal eine Form hat, die zumindest teilweise durch den Kommunikationsrückstau auf dem Bus 34 zu einem bestimmten Zeitpunkt bestimmt ist. Daher ist ein Diagnosesignal, das unter Anwendung asynchroner Kommunikationen gesendet wird, nicht strikt deterministisch und ist daher eventuell nicht sehr effektiv bei der Durchführung einer Diagnose an einem Gerät oder einem Prozeß. Ferner hat der Hauptrechner 12 keine Möglichkeit zu garantieren, daß die von dem Trendobjekt (den Trendobjekten) gesammelten Rückführungsdaten nicht aufgrund von Überschreibvorgängen usw. verlorengehen. Auch hat der Hauptrechner 12 keine Möglichkeit, den Modus der anderen Funktionsblöcke in der Schleife LOOP1 wie etwa des PID-Funktionsblocks 64 zu steuern, ohne den Modus dieses Blocks spezifisch zu ändern.

Bisher mußte ein Benutzer, um in einem Prozeß eine komplette und strikt deterministische Diagnose sicherzustellen, das Prozeßsteuerungsnetz 10 vom Netz nehmen und die Kommunikationsschnittstellen darin neu konfigurieren, so daß der Hauptrechner 12 imstande war, mit Hilfe von synchronen Kommunikationen Sollwertänderungen zu den entsprechenden Geräten zu senden und von entsprechenden Geräten gemessene Daten zu empfangen. Wie oben gesagt wird, wird jedoch bei dieser Vorgehensweise der Prozeß abgeschaltet, und ein Bediener muß immer dann, wenn Diagnosen auszuführen sind, das Prozeßsteuerungsnetz neu konfigurieren; beides ist unerwünscht. Außerdem ist die von dem Hauptrechner 12 während dieses Diagnosevorgangs implementierte Steuerung von der Steuerung verschieden, die von den kommunikativ gekoppelten Funktionsblöcken im Normalbetrieb des Prozesses implementiert wird, und daher kann es sein, daß gesammelte Prozeßdaten die Operation des Prozesses nicht anzeigen, während der Prozeß auf die normale Weise gesteuert wird.

Um diese Nachteile beispielsweise in einem Fieldbus-Prozeßsteuerungsnetz zu überwinden, ist in einem Feldgerät ein Geräte- oder Prozeßdiagnoseablauf gespeichert und von

dem Gerät implementiert und kann dazu dienen, Geräte- und/oder Prozeßdiagnosen an diesem Gerät oder unter Verwendung dieses Geräts auszuführen. Der Diagnoseablauf, der als ein Funktionsblock implementiert sein kann, ist so konfiguriert, daß er mit Funktionsblöcken oder anderen Komponenten des Geräts, in dem er sich befindet, kommuniziert und Daten wie etwa Meßwerte von Geräteparametern oder andere Prozeßparameter beispielsweise auf dem Bus 34 unter Anwendung von synchronen periodischen Kommunikationen (z. B. der Veröffentlichter/Teilnehmer-VCR des Fieldbusprotokolls) empfängt. Auf diese Weise kann die Diagnoseroutine das Gerät, in dem sie sich befindet, deterministisch steuern und Daten auf periodische Weise empfangen und speichern, die einen Geräte- oder einen Prozeßparameter betreffen.

Fig. 6 zeigt als schematisches Blockbild das digitale Feldgerät 16 (von Fig. 3), das eine schleifengespeiste, digital kommunizierende Zweidraht-Zweirichtungs-Kombination aus Positionierer/Ventil ist. Das digitale Feldgerät 16 umfaßt eine Feldgerätesteuereinheit 102, einen I/P-Wandler 104, ein pneumatisches Relais 106, einen Betätiger 108 und ein Ventil 109, die durch verschiedene pneumatische und elektrische Leitungen miteinander verbunden sind.

Das Feldgerät 16 empfängt Betriebssignale und überträgt Statusinformation und Daten in digitaler Form auf dem Zweidraht-Bussegment 34b, bevorzugt entsprechend dem Fieldbus-Standard, und ist daher ein Zweidraht-Positionierer. Ebenso empfängt das Feldgerät 16 Energie, hauptsächlich zum Treiben der Gerätesteuereinheit 102 und des I/P-Wandlers 104, über das kontinuierliche Zweidraht-Schleifenbussegment 34b und ist daher ein schleifengespeistes Gerät.

Wie Fig. 6 zeigt, ist der I/P-Wandler 104 mit der Gerätesteuereinheit 102 elektrisch über eine I/P-Wandlersteuerleitung 110 verbunden und kommuniziert bei der gezeigten Ausführungsform mit der Gerätesteuereinheit 102 unter Anwendung von analogen Steuersignalen.

Der I/P-Wandler 104 erzeugt ein pneumatisches Signal, das die Betätigung des Ventils 109 bewirkt und in elektromechanischen Geräten äußerst nützlich ist, um elektrische Signale in Luftdruck für einen pneumatischen Positionierer umzuwandeln. Der Betätiger 108 bestimmt die Position eines Ventilelements 114 (das ein Ventilschaft sein kann)

des Ventils 109, während ein Positionssensor 116 die Position des Ventilelements 114 erfaßt und ein Rückführungssignal erzeugt, das an die Gerätesteuerereinheit 102 auf einer Signalleitung 117 übermittelt wird. Dieses Positionssignal kann von der Gerätesteuerereinheit 102 genutzt werden, um den Betrieb des Feldgeräts 16 zu steuern, so daß der I/P-Wandler 104 den pneumatischen Druck auf eine Weise treibt, der das Ventilelement 114 veranlaßt, in einer gewünschten Position zu sein. Positions- und andere Rückführungsinformationen können in einer Speichereinheit oder einem Speicher der Gerätesteuerereinheit 102 gespeichert sein, und Zugriff darauf kann von außen über den Bus 34 erfolgen.

Wie es üblich ist, empfängt das Feldgerät 16 eine Druckluftmenge von einer äußeren Quelle (nicht gezeigt) über eine Pneumatikleitung 118, die mit dem I/P-Wandler 104 und dem pneumatischen Relais 106 verbunden ist. Ein Eingangssensor 120, der typischerweise zwischen der äußeren Luftquelle und dem I/P-Wandler 104 angeordnet ist, mißt den eingeführten Speisedruck in der Pneumatikleitung 118 und gibt seinen Meßwert an die Gerätesteuerereinheit 102 ab. Der I/P-Wandler 104 ist mit dem pneumatischen Relais 106 über eine pneumatische Steuerleitung 122 verbunden, und ein I/P-Sensor 124 ist zwischen dem I/P-Wandler 104 und dem pneumatischen Relais 106 positioniert, um den Speiseluftdruck in der Leitung 122 zu messen. Ebenso ist das pneumatische Relais 106 mit dem Betätiger 108 über eine pneumatische Betätigungsleitung 126 verbunden, und ein Relaissensor 128 ist zwischen dem pneumatischen Relais 106 und dem Betätiger 108 positioniert, um den Speiseluftdruck in der Leitung 126 zu messen. Die Pneumatikleitungen 118, 122 und 126 werden als Teile einer einzigen Pneumatikleitung angesehen, die den Wandler 104 mit dem Ventil 109 verbindet.

Im Betrieb steuert die Gerätesteuerereinheit 102 die Betätigung des Ventils 109 durch Steuerung des I/P-Wandlers 104 derart, daß in der pneumatischen Steuerleitung 126 ein gesteuerter Ventilbetätigungsdruck vorgegeben wird. Die Gerätesteuerereinheit 102 sendet ein Steuersignal an den I/P-Wandler 104 über die I/P-Wandlersteuerleitung 110, um einen Ausgangsdruck der Kombination aus dem I/P-Wandler 104 und dem Relais 106 so zu steuern, daß er zwischen ungefähr 0,21 bis 7,06 kscm (3 bis 100 psi) liegt, der einem Steuereingang des Betätigers 108 zugeführt wird. Der Betätiger 108 erzeugt einen Ausgangsdruck, der angelegt wird, um das Ventil 109 zu betätigen.

Wie bekannt ist, wandelt daher der I/P-Wandler 104 elektrische Signale in ein pneumatisches Luftdrucksignal um. Ein Beispiel eines geeigneten I/P-Wandlers 104 ist in der US-PS 5 439 021 mit dem Titel "Electro-Pneumatic Converter", erteilt für B.J. Burlage et al. am 8. August 1995, beschrieben. Ebenso wird das pneumatische Relais 106, das als pneumatischer Verstärker dient, von dem I/P-Wandler 104 nach Maßgabe der Gerätesteuereinheit 102 gesteuert, um den Luftdruck der pneumatischen Betätigungssignalleitung 126 um einen gesteuerten Betrag zu erhöhen. Allgemein gesagt liefert also das pneumatische Relais 106 einen gesteuerten Ausgangsdruck an eine Last oder eine Nutzeinrichtung wie etwa einen Betätiger oder einen Kolben in Abhängigkeit von einem Steuersignal von der Gerätesteuereinheit 102. Ein geeignetes Relais ist in der US-PS 4 974 625 mit dem Titel "Four Mode Pneumatic Relay", erteilt für S. B. Paullus et al. am 4. Dezember 1990, beschrieben. Bei der gezeigten Ausführungsform ist das Relais 106 ein multifunktionales Viermoden-Pneumatikrelais, das für jede Kombination von Direkt-/Umschalt-, Direkt-/Proportional-, Umkehr-/Umschalt- oder Umkehr-/Proportional-Betrieb konfigurierbar ist. Im proportionalen Modus entwickelt das pneumatische Relais 106 einen Druckausgang, der zu einem Druck- oder Krafteingang proportional ist. In einem Ein-/Aus- oder Umschaltmodus erzeugt das pneumatische Relais 106 einen konstanten Druckausgang, der gewöhnlich gleich dem Druck der angelegten Quelle ist, in Abhängigkeit von der Anwendung eines definierten Bereichs von Kraft- oder Drucksteuereingängen. In einem Direktmodus erhöht sich der Ausgangsdruck des pneumatischen Relais 106 mit einem zunehmenden Eingangssignal. In einem Umkehrmodus nimmt der Relaisausgangsdruck mit zunehmendem Eingangssignal ab.

Der Eingangssensor 120, der I/P-Sensor 124 und der Relaisensor 128 sind Druckwandler, die einen Druck-Strom-Wandler zur Umwandlung eines Drucksignals in ein elektrisches Signal enthalten und Rückführungssignale an die Gerätesteuereinheit 102 auf einer Leitung 130 liefern. Der I/P-Sensor 124 ist nützlich für die Diagnose, um einen Ausfall entweder des I/P-Wandlers 104 oder des pneumatischen Relais 106 zu detektieren und beispielsweise zu bestimmen, ob ein Ausfall ein mechanischer Ausfall oder ein elektrischer Ausfall ist. Der I/P-Sensor 124 ist ferner brauchbar zum Detektieren von einigen Systemproblemen einschließlich einer Bestimmung, ob der Eingangsluftdruck zu dem digitalen Feldgerät 16 ausreichend ist. Infolgedessen erlaubt der I/P-Sensor 124 eine rasche Diagnose des Status des I/P-Wandlers 104 und des pneumatischen Relais 106, so daß diese Geräte erforderlichenfalls rasch ausgewechselt werden können.

Bei einer Ausführungsform ist ein geeignetes Ventil 109 zum Gebrauch in dem digitalen Feldgerät 16 eine Ventil- und Betätigereinheit und verwendet einen Feder-/Membranbetätiger an einem Gleitspindelventil, das in einem analogen Gerät verwendet wird, das in der US-PS 4 976 144 mit dem Titel "Diagnostic Apparatus and Method for Fluid Control Valves", erteilt am 11. Dezember 1990 für W.F. Fitzgerald, beschrieben wird. Bei dieser beispielhaften Ausführungsform wird dem Betätiger 108 ein Drucksignal von ungefähr 0,21 kscm (3 psi) als Reaktion auf ein ungefähr 4-mA-Signal zugeführt, das von der Gerätesteueereinheit 102 an den I/P-Wandler 104 angelegt wird, was in einem entsprechenden Druck in der pneumatischen Betätigungssignalleitung 126 resultiert, der nicht ausreichend ist, um das Ventil 109 aus einer vollständig geöffneten Position zu bewegen. Wenn die Feldgerätesteueereinheit 102 den an den I/P-Wandler 104 angelegten Steuerstrom auf ungefähr 20 mA ändert, erzeugt der I/P-Wandler 104 einen Druck in der pneumatischen Betätigungsleitung 126 von ungefähr 1,06 kscm (15 psi), der das Ventil 109 in eine vollständig geschlossene Position drückt. Verschiedene Positionen des Ventils 109 zwischen der vollständig geöffneten Position und der vollständig geschlossenen Position werden durch den Betrieb der Gerätesteueereinheit 102 erreicht, die den an den I/P-Wandler 104 angelegten Eingangsstrom in dem Bereich zwischen 4 mA und 20 mA steuert.

Die Gerätesteueereinheit 102 führt digitale Kommunikationen mit relativ hoher Geschwindigkeit durch, um Steuersignale zu empfangen und Positions- und Druckinformation an einen externen Prozessor oder eine Workstation in dem Prozeßsteuerungsnetz 10 über den Bus 34 zu übertragen. Die Gerätesteueereinheit 102 enthält einen Speicher zur Speicherung der Ergebnisse einer Vielzahl von Diagnoseprüfungen, so daß relevante Daten für die Analyse verfügbar sind. Diagnosevorgänge wie etwa Gerätediagnosen sind im allgemeinen in Form von Softwareprogrammcodes und werden typischerweise in der Gerätesteueereinheit 102 des Feldgeräts 16 codiert, gespeichert und ausgeführt in Kombination mit Programmcodes, die in einer externen Vorrichtung wie etwa einem Prozessor oder der Hauptworkstation 12 ablaufen.

Eine Gerätediagnoseauswertung des Ventils 109 kann durch die Wirkung der Gerätesteueereinheit 102 erfolgen, die den an den I/P-Wandler 104 angelegten Eingangsstrom in einem Bereich steuert, der zur Prüfung des Ventils 109 zwischen der vollstän-

dig geöffneten und der vollständig geschlossenen Position ausreichend ist. Während der Gerätediagnoseauswertung werden die Ausgänge des Eingangssensors 120, des I/P-Sensors 124 und des Relaisensors 128 von der Gerätesteuereinheit 102 überwacht zur Erfassung der pneumatischen Drücke in den Pneumatikleitungen 118, 122 bzw.

126, die für die Analyse verwendet werden. Der Ausgang des Positionssensors 116 wird ebenfalls überwacht, um die Position oder Bewegung des Ventilschafts 114 zu erfassen, die einer Position oder einer Bewegung eines Ventilstopfens (nicht gezeigt) innerhalb des Ventils 109 entspricht.

Daher wird ein Prüfbetriebszyklus des Ventils 109 unter Steuerung durch die Gerätesteuereinheit 102 durchgeführt, indem ein gesteuerter variabler Strom an den I/P-Wandler 104 angelegt, der Druck in den Pneumatikleitungen 118, 122 und 126 erfaßt und die Position des Ventilschafts 114 unter Verwendung des Positionssensors 116 erfaßt wird. Auf diese Weise empfängt die Gerätesteuereinheit 102 gleichzeitig zeitlich veränderliche elektrische Signale, die die Drücke an den gezeigten Stellen und die Position des Ventils 109 bezeichnen, und kann diese Signale dazu nutzen, jede Art von Diagnoseparametern des Geräts auf jede bekannte oder gewünschte Weise zu bestimmen.

Herkömmliche Feldgeräte weisen charakteristisch keinen Positionssensor wie etwa den Sensor 116 auf und verwenden keine Positionssensorergebnisse für diagnostische Auswertungen. Ferner enthalten herkömmliche Feldgeräte keine Sensoren wie etwa den Eingangssensor 120, den I/P-Sensor 124 und den Relaisensor 128 zur Druckmessung bei der pneumatischen Steuerung und zur Umwandlung des Drucksignals in ein elektrisches Signal, um die diagnostische Auswertung zu vereinfachen. Diese Sensoren und insbesondere der I/P-Sensor 124 verbessern jedoch die diagnostischen Fähigkeiten durch Vereinfachung der Lokalisierung von Ausfall- oder Fehlerzuständen in dem Feldgerät 16. Insbesondere unterstützt der I/P-Sensor 124 die Differenzierung zwischen Ausfällen des Ventils 109, anderen Ausfällen in dem Feldgerät 16 und Ausfällen außerhalb des Feldgeräts 16 einschließlich Ausfällen der Pneumatikleitung, die das Feldgerät 16 speist. Der I/P-Sensor 124 ist außerdem nützlich bei der Durchführung einer Diagnoseprüfung zur Bestimmung des Betriebsstatus des I/P-Wandlers 104, des pneumatischen Relais 106, des Feldgeräts 16 und des Prozeßsteuerungssystems 10 im allgemeinen. Bei einer Ausführungsform werden der I/P-Wandler 104 und das pneumatische

Relais 106 unter Anwendung eines Diagnoseprüfverfahrens geprüft, bei dem der I/P-Wandler 104 in den vollständigen Offenzustand getrieben wird, um den vollständigen Luftdruck zu messen, der dem Ventil 109 zugeführt wird. Während der I/P-Wandler 104 in den Offenzustand getrieben wird, mißt der I/P-Sensor 124 ständig den Druck in der pneumatischen Steuerleitung 122. Wenn der Druck geringer zu werden beginnt, zeigt die Prüfung an, daß die Luftzufuhr ungenügend sein kann. Eine weitere Diagnoseprüfung der ausreichenden Luftzufuhr erfolgt durch Pumpbetätigen des Ventils 109, indem ein oszillierendes Signal an den I/P-Wandler 104 angelegt wird, so daß das Ventil 109 eine Saugwirkung in bezug auf die Luftzufuhr beginnt, woraufhin unter Verwendung des I/P-Sensors 124 die maximalen Durchfluß- und maximalen Druckwerte gemessen werden.

Wie Fig. 7 zeigt, umfaßt die Gerätesteereinheit 102 einen Mikroprozessor 140, eine Schnittstelle 142, eine Bustrennschaltung 144, eine Vielzahl von Speichereinrichtungen wie einen RAM 146, einen ROM 148 und einen nichtflüchtigen RAM bzw. NVRAM 150 und eine Vielzahl von Signalverarbeitungseinrichtungen wie einen A/D-Wandler 152, einen D/A-Wandler 154 und einen Multiplexer 156. Die Schnittstelle 142 (die ein Busverbinder ist) ist eine Schaltung, die eine Serien-Parallel-Protokollumwandlung und eine Parallel-Seriell-Protokollumwandlung ausführt und dazu dient, Datenpaketen entsprechend jeder gewünschten Protokolldefinition wie etwa dem Fieldbusprotokoll Rahmeninformation hinzuzufügen. Die Bustrennschaltung 144 ist eine Schaltung, die dazu dient, ein Zweidrahtmedium-Kommunikationssignal auf dem Bus 34 in eine digitale Darstellung des Kommunikationssignals umzuwandeln, und liefert von dem Bus 34 erhaltene Energie an andere Schaltungen in der Gerätesteereinheit 102 sowie an den I/P-Wandler 104. Die Bustrennschaltung 144 kann außerdem eine Wellenformung und Zeichengebung auf dem Bus 34 ausführen.

Der A/D-Wandler 152 ist mit Wandlern wie etwa den Positions- und Druckwandlern des Positionssensors 116 und der Drucksensoren 120, 124 und 128 von Fig. 6 sowie mit allen anderen gewünschten analogen Eingabeeinrichtungen verbunden. Der A/D-Wandler 152 hat zwar eventuell eine begrenzte Anzahl von Eingangskanälen, aber der Multiplexer 156 kann verwendet werden, um die Abtastung einer Vielzahl von Signalen zu ermöglichen. Falls gewünscht, kann der Multiplexer 156 ein Feld von Verstärkern aufweisen, die zwischen die Signalleitungen 117 und 130 (Fig. 6) geschaltet sind, um

die dorthin übertragenen Positions-, Druck- und sonstigen Rückführungssignale zu verstärken. Der D/A-Wandler 154 führt die Digital-Analog-Umwandlung an Signalen durch, die von dem Mikroprozessor 140 entwickelt werden, um analogen Komponenten wie etwa dem I/P-Wandler 104 zugeführt zu werden.

5

Bei einer typischen Diagnoseprüfungsanwendung erzeugt die Steuereinheit 102 ein Ausgangsprüfsignal von 0 bis 30 mA an den I/P-Wandler 104 beispielsweise in programmierter Rampenform, schrittweiser Änderungsform, Ein/Aus-Form (oder jeder anderen gewünschten Weise) zur Betätigung des Ventils 109 über einen vorbestimmten Bereich von pneumatischen Drücken und empfängt Diagnoseprüf Ausgangssignale, die von dem Druckeingangssensor 120, dem I/P-Sensor 124, dem Relaisensor 128 und dem Positionssensor 116 entwickelt werden. Bestimmte Prüfvorgänge können an dem Feldgerät 16 angegeben werden und werden im allgemeinen außerhalb des Feldgeräts 16 ausgelöst unter Verwendung einer Ein-/Ausgabeeinrichtung wie etwa einer Workstation, obwohl die Vorgänge (und ihnen zugeordnete erforderliche Information) auch direkt unter Verwendung einer Eingabeeinrichtung wie etwa einer Tastatur in die Steuereinheit 102 eingegeben werden können. Falls gewünscht, können die Prüfvorgänge jedoch in der Steuereinheit 102 gespeichert sein. Ebenso wird Diagnoseprüfergebnisinformation, die von der Feldgerätesteuereinheit 102 gesammelt oder erzeugt wird, typischerweise an eine externe Ein-/Ausgabeeinrichtung übertragen, aber diese Information kann auch von der Gerätesteuereinheit 102 beispielsweise unter Verwendung eines CRT-Displays oder eines Druckers direkt angezeigt werden.

10

15

20

25

30

Das Feldgerät 16 führt Diagnoseprüfoperationen wie Geräte- und Prozeßdiagnosen aus unter Verwendung eines Programmierspracheninterpretierers, der in die Steuereinheit 102 eingebettet ist und Befehle wie diejenigen interpretiert, die die Durchführung von Diagnoseverfahrensschritten verlangen. Der Spracheninterpretierer ist bevorzugt in einem in dem PROM 148 gespeicherten Programmcode implementiert und wird in dem Mikroprozessor 140 ausgeführt. Bei einigen Ausführungsformen ist eine Prüfdefinition (Diagnosedefinition) oder -prozedur in dem PROM 148 codiert und somit vorher in das Feldgerät 16 geladen. Bei anderen Ausführungsformen wird eine Prüfdefinition (Diagnosedefinition) oder -prozedur zu oder vor dem Zeitpunkt heruntergeladen, zu dem die Prozedur ablaufen soll, und wird beispielsweise in dem RAM 146 zur Ausführung durch den Mikroprozessor 140 gespeichert. Bei einer typischen Ausführungsform sind einige

Diagnosefunktionen in dem PROM 148 hartcodiert und andere Funktionen werden heruntergeladen, was die Auslegung und Implementierung von neuen Diagnoseprüfungen ohne Modifikation der permanenten oder hartcodierten Software in dem Gerät 16 erlaubt. Der Spracheninterpretierer wird zwar im Zusammenhang mit der Durchführung von Prozeß- oder Gerätediagnosen in einem Gerät vom Fieldbus-Typ (etwa einem Fieldbus-Ventil) beschrieben, der Spracheninterpretierer kann jedoch in jeder Art von eingebetteter Steuereinheit implementiert sein, was die Nutzung von gemeinsamen Diagnoseprüfoperationen in jedem anderen Typ von eingebetteter Steuereinheit ermöglicht.

Im Betrieb empfängt das Feldgerät 16 Anweisungen über den Bus 34 von einer Bedienerkonsole oder Hauptworkstation 12 in dem Prozeßsteuerungsnetz 10. Der Spracheninterpretierer, der in der Gerätesteuereinheit 102 abläuft, interpretiert die Anweisungen und führt die durch die Anweisungen definierten Operationen aus. Bei einer gezeigten Ausführungsform ist die Sprachendefinition in einer in TABELLE 1 gezeigten Anwenderschnittstellen-Befehlstabelle wie folgt dargestellt:

TABELLE 1	
Befehl	Parameter
Absolutbewegung	Position
Relativbewegung	Verschiebung
Pause	Zeit
Inkrement vorgeben	Inkrement
Aufwärtsinkrement	
Abwärtsinkrement	
Schleifenstart	Anzahl von Iterationen
Schleifenende	
Stopp	
Datenrate	ganzzahliges Vielfaches von Servorate
//Kommentarzeile	
Etikett:	
Subroutine aufrufen	Bezeichnung der Subroutine
Rücksprung	
Prüfvariable	Variable, Wert, Adresse

In dieser Spezifikation der Diagnosesprache kann ein Benutzer einen Befehlsnamen verwenden, um den Befehl für die Ausführung zu bezeichnen. Eine Kommentarzeile ist jede Zeile, die mit zwei Schrägstrichen (//) entsprechend der Standardkonvention C/C++ beginnt. Ein Etikett ist jedes Wort, auf das ein Punkt folgt.

Eine Diagnoseprüfprozedur kann in einer Bedienerkonsole wie etwa in dem Hauptrechner 12 definiert sein, der eine Folge von Anweisungen in Übereinstimmung mit der Sprachendefinition erzeugt, die dazu bestimmt ist, die Diagnoseprüfprozedur zu implementieren. Die Bedienerkonsole überträgt dann die Folge von Anweisungen, die in der interpretierten Sprache codiert sind, über den Bus 34 zu dem digitalen Feldgerät 16 beispielsweise unter Verwendung von asynchronen Kommunikationen in dem Feldbusprotokoll. Der Spracheninterpretierer, der in der Gerätesteuereinheit 102 ausgeführt wird, speichert die empfangenen Anweisungen und interpretiert die Anweisungen sequentiell zur Steuerung des Ventils 109 entsprechend den Anweisungen, um so die Diagnoseprüfung auszuführen. Diagnoseprüfprozeduren können beispielsweise das Feldgerät 16 so steuern, daß das Ventil 109 wiederholt schrittweise bewegt, das Ventil 109 aufwärts oder abwärts bewegt, das Ventil 109 um eine ausgewählte Strecke in einer ausgewählten Richtung bewegt wird usw. Die Diagnoseprüfprozedur steuert ferner das Sammeln von Daten von Sensoren in dem Feldgerät 16 (sowie von anderen Geräten) und steuert die Übertragung von Daten an die Steuerungskonsole oder Hauptworkstation 12 über den Bus 34. Falls gewünscht, können die Diagnoseprüfprozeduren, die durch an die Steuereinheit 102 gelieferte Anweisungen implementiert werden, natürlich auch die empfangenen Daten verarbeiten, um Diagnoseergebnisse zu ermitteln, und können diese Ergebnisse an den Hauptrechner 12 oder eine andere Displäeinrichtung senden.

Somit steuert der Diagnosespracheninterpretierer innerhalb der Gerätesteuereinheit 102 die Operation des Feldgeräts 16 nach Maßgabe von programmierten Anweisungen, so daß Diagnoseprüfprozeduren außerhalb des digitalen Feldgeräts 16 definiert werden können, damit Diagnoseprüfungen ohne Modifikation des Feldgeräts 16 frei definiert und modifiziert werden können. Ebenso können neue Diagnosesteuerungsprozeduren entwickelt und zu dem Feldgerät 16 übermittelt werden, nachdem das Feldgerät 16 in dem Prozeßsteuerungsnetz 10 installiert worden ist. Falls gewünscht, kann jedoch das

Gerät 16 auch oder alternativ Geräte- und/oder Prozeßdiagnoseprüfanweisungen implementieren, die in dem Gerät bei der Herstellung oder zu einem anderen Zeitpunkt gespeichert wurden.

5 Eine Steuerkonsole (wie etwa der Hauptrechner 12) umfaßt typischerweise Diagnoseentwicklungswerkzeuge wie Spracheditoren und Simulatoren zur Entwicklung der Steuerroutrinen in der Diagnosesprache für die Ausführung durch das Feldgerät 16. Die Steuerkonsole umfaßt außerdem typischerweise Analysewerkzeuge für die Analyse von Daten, die von dem Feldgerät 16 auf dem Bus 34 empfangen werden.

10 Der Vollständigkeit halber sind beispielhafte Diagnoseprogrammcodes in einer interpretierenden Sprache wie folgt für die Steuerung des Ventils 109 dargestellt:

Programmcodes

(1) SAMA Statische Testdefinition

15 //Zyklus 0 BIS 100% DREIMAL
DatenRate 1
ZYKLUS: Loop 3
20 AbsolutBewegung 0.0
Pause 10000
AbsolutBewegung 100.0
Pause 10000
LoopEnde
25 //BEWEGEN BIS 50 % VIERMAL AUFWÄRTSSCHRITTE
AbsolutBewegung 50.0
Pause 10000
InkrementSetzen 10.0
AUFWÄRTS: LOOP 4
30 InkrementAufwärts
Pause 10000
LoopEnde
//SCHRITTWEISE ABWÄRTS ACHTMAL
ABWÄRTS: Loop 8
35 InkrementAbwärts
//SCHRITTWEISE AUFWÄRTS VIERMAL
AUF2: Loop 4
InkrementAufwärts
Pause 10000
40 LoopEnde
Stopp

(2) Schritttänderungs-Testdefinition

DatenRate 1
AbsolutBewegung 50.0
Pause 10000
5 AbsolutBewegung 60.0
Pause 10000
Stopp

(3) Stufenrampen-Testdefinition

10 DatenRate 1
Absolutbewegung 50.0
InkrementSetzen 0.5
//AUFWÄRTSSCHRITTE UM 0.5 ZEHNMAL
15 AUFWÄRTS1: Loop 10
InkrementAUFWÄRTS
Pause 1000
LoopEnde
//ABWÄRTSSCHRITTE UM 0.5 ZEHNMAL
20 ABWÄRTS1: Loop 10
InkrementAbwärts
Pause 1000
LoopEnde
InkrementSetzen 1.0
25 //AUFWÄRTSSCHRITTE UM 0.5 ZEHNMAL
AUFWÄRTS2: Loop 10
InkrementAufwärts
Pause 1000
LoopEnde
30 //ABWÄRTSSCHRITTE UM 0.5 ZEHNMAL
ABWÄRTS2: Loop 10
InkrementAbwärts
Pause 1000
LoopEnde
35 InkrementSetzen 2.0
//AUFWÄRTSSCHRITTE UM 0.5 ZEHNMAL
AUFWÄRTS3: Loop 10
InkrementAufwärts
Pause 1000
40 LoopEnde
//ABWÄRTSSCHRITTE UM 0.5 ZEHNMAL
ABWÄRTS3: Loop 10
InkrementAbwärts
Pause 1000
45 LoopEnde
Stopp

(4) Schrituntersuchungs-Testdefinition

50 DatenRate 1

//SCHRITT AUF, AB, AB, AUF, DANN SCHRITTGRÖSSE INKREMENTIEREN UND
//WIEDERHOLEN, BIS ÄNDERUNGEN DETEKTIERT WERDEN.

InkrementSetzen 0.5

InkrementAufwärts

5 Pause 100

InkrementAbwärts

Pause 100

10 InkrementAbwärts

Pause 100

InkrementAufwärts

Pause 100

InkrementSetzen 1.0

15 InkrementAufwärts

Pause 100

InkrementAbwärts

Pause 100

InkrementAbwärts

20 Pause 100

InkrementAufwärts

Pause 100

InkrementSetzen 2.0

InkrementAufwärts

25 Pause 100

InkrementAbwärts

Pause 100

InkrementAbwärts

Pause 100

30 InkrementAufwärts

Pause 100

InkrementSetzen 5.0

InkrementAufwärts

Pause 100

35 InkrementAbwärts

Pause 100

InkrementAbwärts

Pause 100

InkrementAufwärts

40 Pause 100

Stopp

Die erste oben angegebene Prüfung (1) durchläuft drei Zyklen, führt vier Schritte aus der halbgeöffneten Position aus, geht achtmal abwärts und viermal aufwärts. Die zweite Prüfung (2) führt eine Schrittländerung von 50 auf 60 % der Absolutposition des Ventils aus. Die dritte Prüfung (3) führt drei Zyklen einer stufenförmigen Rampenwellenform aus, beginnend bei 50 % der Absolutposition des Ventils. Die vierte Prüfung (4) wiederholt eine Serie von Schritten mit ansteigenden Werten, bis eine Änderung im Ventil detektiert wird.

In diesen Routinen implementiert die Feldgerätesteuereinheit 102 eine bedingte Pause, um die Aufzeichnung zu stoppen, und setzt ein Bit in einen Speicherplatz, um anzuzeigen, wo eine Prüfung gestoppt wird. Die Steuereinheit 102 des Feldgeräts implementiert auch ein Branch/GOTO-Statement, ein Loop-forever-Statement und einen Zwangstopp, wenn ein Außer-Betrieb-Flag gesetzt ist. Die Pausen sind mit Servolaufzeiten synchronisiert, so daß die Prüfung nicht außer Synchronisation mit dem Ventil gelangt.

Der beispielhafte Programmcode zeigt nur einige Beispiele der Arten von Diagnoseprüfungen, die von dem Feldgerät 16 durchgeführt werden können, es gibt aber viele andere Diagnoseprüfungen, die durch Programmanweisungen durchgeführt werden können, die dem Feldgerät 16 zugeführt werden und beispielsweise umfassen: eine statische Zyklusprüfung, bei der das Ventil 109 10 % aufwärts, 10 % abwärts, 10 % aufwärts, 10 % abwärts usw. über eine Mehrzahl von Zyklen bewegt wird. Ebenso können alle Gerätediagnosemessungen durchgeführt werden, die beispielsweise umfassen: einfache Messungen des Ventilweges oder der Drücke innerhalb des Geräts 16, entwickelt von den Sensoren 116, 120, 124 und 128, und/oder alle gewünschten Parameter, die von diesen oder anderen Messungen ableitbar sind, wie beispielsweise (1) ein dynamisches Fehlerband, das ein Diagramm des Weges (z. B. des Ausgangs des Positionssensors 116) gegenüber dem Eingang (z. B. dem Steuersignal, das der Steuereinheit 102 zugeführt wird) ist, (2) ein Diagramm des Treibersignals (das das Ausgangssignal der Steuereinheit 102 ist, wie es dem I/P-Wandler 104 zugeführt wird) gegenüber einer Druckmessung, (3) ein Diagramm des Treibersignals gegenüber dem Eingangssignal, (4) Ausgangssignal, das ein Diagramm des Weges gegenüber dem Treibersignal ist, (5) Ventilsignatur, die ein Diagramm des Drucks gegenüber der Strecke ist, usw. Natürlich können die in diesen Prüfungen angegebenen Drucksignale alle gewünschten Drucksignale sein wie etwa diejenigen, die von jedem der Sensoren 120, 124 und/oder 128 gemessen werden.

Die Diagnosesprache und der Diagnosespracheninterpretierer sind zwar vorteilhaft in einem Prozeßsteuerungsnetz 10 implementiert, das Fieldbus-Kommunikationen zur Kommunikation mit einem digitalen Feldgerät 16 in Form eines schleifengespeisten, digital kommunizierenden Zweirichtungs-Zweidraht-Positionierers verwendet, der Spracheninterpretierer kann jedoch in anderen Ausführungsformen implementiert werden.

Beispielsweise kann der Diagnosespracheninterpretierer in jeder eingebetteten Steuereinheit implementiert sein, die unter Verwendung einer gewünschten Kommunikationstechnologie wie etwa digital, analog, optisch u. dgl. kommuniziert. Der gezeigte Diagnosespracheninterpretierer kommuniziert zwar nach dem Fieldbus-Standardprotokoll, alternative Ausführungsformen des Diagnosespracheninterpretierers können aber in einer eingebetteten Steuereinheit, die unter Verwendung anderer Kommunikationsprotokolle wie beispielsweise des HART-, Profibus-Protokolls usw. kommuniziert, und in Systemen, die andere als Zweidrahtbusse verwenden, etwa Systemen, die Vierdrahtbusse verwenden, implementiert sein. Ebenso kann der Diagnosespracheninterpretierer in anderen Ventilbauarten implementiert und verwendet werden, beispielsweise in elektronischen und hydraulischen Ventilen sowie zusätzlich zu Ventileinrichtungen in anderen Arten von Geräten.

Die Diagnosesprache und der Diagnosespracheninterpretierer wurden zwar so beschrieben, daß sie eine bestimmte Anweisungsgruppe definieren, aber andere Anweisungsgruppen können entsprechend den Spezifikationen der eingebetteten Steuereinheit, innerhalb welcher die Sprache und der Interpretierer definiert sind, implementiert werden.

Natürlich können sowohl Geräte- als auch Prozeßdiagnoseprüfungen unter Abgabe eines Befehls, der eine oder mehrere spezielle Diagnoseprüfprozeduren verlangt, an einer Bedienerkonsole wie etwa der Hauptworkstation 12 durchgeführt werden. Bei der gezeigten Ausführungsform sind die Diagnoseprüfprozeduren in zwei Softwareprogrammcodes implementiert. Ein erster Code führt in einem Prozessor aus, der außerhalb des Feldgeräts 16 liegt, beispielsweise in der Hauptworkstation 12, um eine Diagnoseprüfung zu erzeugen oder auszulösen und um gesammelte Daten zu empfangen und sie zu analysieren, während ein zweiter Code in der Gerätesteuereinheit 102 ausführt, um eine Diagnoseprüfung zu implementieren, die darin gespeichert ist oder von dem Hauptrechner 12 in Form von Programmanweisungen geliefert wird. Dagegen werden Diagnosen in einem herkömmlichen Steuerungssystemnetz ausschließlich durch Software ausgeführt, die in einem Prozessor einer Steuerkonsole abläuft. Viele Vorteile werden erhalten durch Ausführen von Diagnoseprüfungen auf der Feldgeräteebene anstatt auf der Ebene einer Steuerkonsole. Beispielsweise können Diagnoseprüfungen parallel durchgeführt und auf viele Feldgeräte verteilt werden, indem diese Prüfungen

auf der Geräteebene durchgeführt werden. Ebenso kann eine genauere Prüfung in Prozeßsteuerungsnetzen mit verteilten Steuerfunktionen wie etwa im Feldbusprotokoll durchgeführt werden, wo ein Hauptrechner eventuell nicht in der Lage ist, den Betrieb des Feldgeräts deterministisch zu steuern, weil der Hauptrechner mit dem Feldgerät zur Durchführung einer Diagnoseprüfung auf asynchrone Weise kommunizieren muß.

Die Implementierung der durch das Feldbusprotokoll definierten digitalen Zweirichtungs-Kommunikation ist äußerst vorteilhaft, um die Diagnosegeschwindigkeit zu steigern, und zwar sowohl durch einen erhöhten Datendurchsatz als auch durch Vereinfachung der parallelen Durchführung von Diagnosen bei einer Vielzahl von Feldgeräten. Durch die Verwendung des Feldbusprotokolls werden regelmäßig geplante Nachrichten zu vorher festgelegten Zeiten übertragen, und ungeplante Nachrichten einschließlich Diagnosenachrichten und -daten, Kalibrierinformation und sonstige Informationen wie Statusanzeigen werden übertragen, wenn die Nachrichten oder Daten bereit sind und der Bus 34 des Feldgeräts nicht anderweitig belegt ist. Diagnoseanforderungsnachrichten werden von Ziel-Feldgeräten empfangen, und Diagnoseprüfungen werden von den Feldgeräten in bezug auf Operationen anderer Feldgeräte asynchron durchgeführt. Wenn ein Diagnoseprüfvorgang komplett ist, liefert ein Feldgerät eine Antwort wie etwa Ergebnisdaten, wenn der Bus 34 des Feldgeräts verfügbar ist. Somit kann, wie oben gesagt wird, eine Vielzahl von Feldgeräten Diagnoseprüfungen parallel ausführen.

Gemäß Fig. 8 zeigt ein Flußdiagramm 200 eine Technik zur Durchführung von Diagnoseprüfungen innerhalb des Feldgeräts 16. In einem Schritt 202 empfängt das Feldgerät 16 eine Anforderung zur Durchführung einer Folge von Anweisungen, die eine oder mehrere Diagnoseprüfungen implementieren. Natürlich kann die Hauptworkstation 12 solche Anforderungen an jedes oder an eine Vielzahl von Feldgeräten gleichzeitig aussenden und jedem Feldgerät erlauben, parallel Diagnosedaten zu sammeln. Wenn eine Vielzahl von Feldgeräten gleichzeitig Prüfungen ausführt, kann die Workstation 12 Daten über einen verlängerten Zeitraum so schnell sammeln, wie die Diagnoseprüfungen durchgeführt und die Ergebnisse auf dem Bus 34 von den einzelnen Feldgeräten verfügbar gemacht werden. Herkömmliche Einrichtungen, die einen speziellen Satz von Leitern zur Kommunikation mit jedem Feldgerät verwenden, haben jeweils nur zu einem Feldgerät Zugang und führen jeweils nur eine einzige Prüfung für das einzelne Feldgerät aus.

In Schritt 204 führt das Feldgerät 16 eine Serie von Anweisungen zur Implementierung der einen oder der mehreren Diagnoseprüfungen entsprechend der Anforderung aus. Selbstverständlich können, wie oben gesagt wird, die Anweisungen in dem Speicher des Feldgeräts 16 gespeichert oder dem Feldgerät beispielsweise mittels asynchroner Kommunikationen von dem Hauptrechner 12 zugeführt werden. Während die Prüfung implementiert wird, werden ein oder mehr Parameter wie der Ventilweg, der Druck usw. parallel gemessen. Während also herkömmliche Feldgeräte typischerweise einen Befehl für eine einzige Diagnoseprüfmessung empfangen, die Messung seriell ausführen und aufgrund der begrenzten Kommunikationsbandbreite und des Mangels an Speicherkapazität in diesen Feldgeräten auf die Anforderungen mit einem einzigen Meßwert antworten, kann das Feldgerät 16 eine Anforderung für eine Vielzahl von Prüfungen unter Anwendung eines flexiblen Prüfprotokolls empfangen, die Vielzahl von Prüfungen ausführen und dann mit den während der Prüfungen gesammelten Ergebnissen antworten.

In einem Schritt 206 speichert das Feldgerät 16 Parameterergebnisse der Vielzahl von Prüfungen, die für jede der Diagnoseprüfungen gemessen wurden, in einem Speicher und überträgt in einem Schritt 208 die Daten an die externe anfordernde Einrichtung. Die digitale Zweidraht-Zweirichtungs-Kommunikation entsprechend dem Fieldbus-Standard verbessert den Prüfergebnisdurchsatz des digitalen Feldgeräts 16 erheblich. Tatsächlich verbessert die digitale Kommunikation unter Verwendung des Fieldbusprotokolls die Datenübertragungszeit ungefähr um das Dreißigfache gegenüber HART-Systemen, so daß dann, wenn das Fieldbusprotokoll zur Durchführung von Diagnoseprüfungen parallel an einer Vielzahl von Feldgeräten verwendet wird, die Diagnoseprüfzeit für ein Prozeßsteuerungsnetz, das viele Feldgeräte aufweist, erheblich verkürzt wird.

Herkömmliche Feldgeräte haben typischerweise ein separates Drahtpaar, das jedes Feldgerät mit einem Netz verbindet, so daß jedes Feldgerät exklusiv Zugang zu den Drähten hat, bei der gezeigten Ausführungsform dagegen werden die Ergebnisse der Diagnoseprüfungen an die Bedienerkonsole oder die Hauptworkstation 12 über den Bus 34 unter Anwendung des Fieldbus-Standardprotokolls übertragen, wodurch die zur

Kommunikation mit dem Hauptrechner 12 erforderliche Menge an Draht verringert wird.

Es versteht sich, daß während einer Diagnoseprüfprozedur der Mikroprozessor 140 den D-A-Wandler 154 so steuert, daß er dem I/P-Wandler 104 ein veränderliches Steuersignal zuführt. Für jeden speziellen Steuersignalwert und jede spezielle Abtastzeit weist der Mikroprozessor 104 den D-A-Wandler 152 an, die druck- und/oder positionsbezogenen elektrischen Signale, die von den Sensoren 116, 120, 124 und 128 entwickelt werden (sowie etwaige weitere Signale von anderen Sensoren), zu messen. Während der Mikroprozessor 104 die Feldgerätesteureinheit 102 durch einen Meßbetriebszyklus leitet, hat die Feldgerätesteureinheit 102 Zugang zu der Druck- und Ventilweginformation, die verarbeitet oder gespeichert wird. Die gesammelten Daten werden häufig vorübergehend in dem RAM 146 gespeichert und einer externen Einrichtung wie etwa einer Hauptworkstation 12 zugeführt, und zwar häufig zur anschließenden Verarbeitung, Analyse und Anzeige. Selbstverständlich kann der Mikroprozessor 140, falls gewünscht, auch eine Analyse durchführen.

Beispielsweise kann von dem Relaisensor 128 gemessene Druckinformation und von dem Positionssensor 116 gemessene Druckinformation gemeinsam analysiert werden, um die Änderung des Ventilmembrandrucks als eine Funktion der Ventilstellung zu bestimmen. Ebenso können Drücke, die von dem Eingangssensor 120, dem I/P-Sensor 124 und dem Relaisensor 128 gemessen werden, jeweils gemeinsam mit der von dem Positionssensor 116 gemessenen Position analysiert werden, um einen Abweichungszyklus zu erzeugen, der eine vollständige Analyse des Betriebs des Ventils 109 einschließlich der Charakteristiken der Linearität, Hysterese und des Bereichs zeigt.

In Fig. 9 zeigt ein Flußdiagramm 300 ein Diagnoseprüfprotokoll, das in dem Feldgerät 16 implementiert ist, um eine Diagnoseprüfung auszuführen. Ein Bediener erzeugt eine Folge von Diagnoseanweisungen an einer Bedienerkonsole wie etwa der Workstation 12 und überträgt die Diagnoseanweisungen an das Feldgerät 16, das diese Anweisungen im Speicher speichert. Alternativ oder zusätzlich können Diagnoseanweisungen während der Herstellung des Geräts in dem Speicher des Feldgeräts 16 gespeichert werden.

In einem Schritt 302 empfängt das Feldgerät 16 einen Diagnosebefehl über den Feldgerätebus 34. Falls gewünscht, kann der Diagnosebefehl in Form von Anweisungen in einer interpretationsfähigen Diagnosesprache sein, die zur Ausführung in der Feldgerätesteuereinheit 102 codiert ist, oder kann eine Anweisung sein, bereits in dem Gerät 16 gespeicherte Anweisungen auszuführen. Die Gerätesteuereinheit 102 führt die verschiedenen Anweisungen aus und veranlaßt Steuerelemente in dem digitalen Feldgerät 16 wie etwa den I/P-Wandler 104 und den Betätiger 108, das Ventil 109 zu manipulieren und Sensoren wie etwa den Eingangssensor 120, den I/P-Sensor 124, den Relais-sensor 128 und den Positionssensor 116 zu veranlassen, Messungen durchzuführen.

Die Steueranweisungen können auch Anweisungen zur Verarbeitungen der Meßwerte und zur Formatierung von Daten zur Darstellung für einen Benutzer aufweisen. Zusätzliche Anweisungen können die Feldgerätesteuereinheit 102 veranlassen, verarbeitete oder formatierte Daten über den Feldgerätebus 34 an die anfordernde Einrichtung zu senden.

In einem Schritt 304 führt das digitale Feldgerät 16 auf Anforderung eine Prüfungsprozedur aus, bei der beispielsweise das Ventil 109 aus einem vollständig geschlossenen Zustand in einen vollständig geöffneten und dann zurück in einen vollständig geschlossenen Zustand bewegt wird. Ebenso kann das digitale Feldgerät 16 auf Anforderung eine Vielzahl von schrittweisen Prüfungen ausführen, beispielsweise eine Einzelschrittbewegung und Analyse und eine Zehnschrittbewegung und Antwortmessung. Eine stufenweise Rampenprüfung kann ebenfalls angewandt werden und umfaßt eine Serie von Schritten, ausgehend von einem geringfügigen Öffnungsgrad bis zu einem großen Öffnungsgrad des Ventils, beispielsweise eine Rampe im Bereich von 10 % bis 90 % in Schritten von beispielsweise 10 %-Inkrementen. Eine abgestufte Untersuchung umfaßt das Öffnen des Ventils in vordefinierten Schritten wie etwa 1 %, 2 %, 5 % und 10 %, wobei das Ventil in einer ersten Richtung um eine bestimmte Schrittgröße bewegt, stabilisiert und dann in einer zweiten Richtung mit der bezeichneten Schrittgröße bewegt wird.

In einem Schritt 306 wird die physische Konfiguration einer Diagnoseprüfung durch Befehle vorgegeben, die dem digitalen Feldgerät 16 übermittelt werden. Die physischen Konfigurationsvariablen umfassen ein Treibersignal, das dem Betätiger 108 zugeführt wird, eine Druckeinstellung, die an den I/P-Wandler 104 angelegt wird, sowie einen

Betätigerdruck- und einen Ventilstreckenmeßwert. Die physischen Konfigurationsvariablen können als unabhängige Prüfvariablen in einigen Diagnoseprüfungen vorgegeben werden und in anderen Prüfungen als abhängige Parameter überwacht werden.

- 5 In einem Schritt 308 mißt das Feldgerät 16 vorbestimmte Parameter für eine bestimmte physische Prüfkonfiguration. Für viele Diagnoseprüfungen wird eine Vielzahl von Parametern für eine einzige Prüfkonfiguration gemessen. Typische Parameter, die unter Anwendung einer einzigen Prüfkonfiguration gemessen werden, umfassen Ventilposition, Prozeßvariablen, Betätigerluftdruck, Versorgungsdruck, Treibersignal, Wandler-
- 10 Sollwert und I/P-Luftdruck, um beispielsweise die Bestimmung der Ventilsignatur, des Ausgangssignals, des dynamischen Fehlerbands, des Antriebs gegenüber dem Druck sowie der Strecke gegenüber der Eingangssignalamplitude zu ermöglichen.

- 15 In einem Schritt 310 speichert das Feldgerät 16 durch die Operation der Gerätesteuereinheit 102 die Vielzahl von Parametern für eine einzige Prüfkonfiguration beispielsweise in dem RAM 146. In einem Schritt 312 werden die Daten an den Hauptrechner 12 oder eine andere Einrichtung übermittelt. Wenn die Prüfung unter Verwendung einer anderen Diagnoseprüfkonfiguration ausgeführt werden soll, geht das Feldgerät 16 durch die Operation der Gerätesteuereinheit 102 in einem Bedingungsschritt 314 in der
- 20 Schleife zu Schritt 306 zurück, um die Prüfkonfiguration zu modifizieren.

- Die Diagnoseprüfungen in dem gezeigten Feldgerät 16 sind gegenüber herkömmlichen Feldgeräten erheblich verbessert, und zwar zum Teil durch Verbesserungen der Struktur des Diagnoseprotokolls und zum Teil durch Verbesserungen in der Kommunikation
- 25 und zum Teil durch die Implementierung zusätzlicher Sensoren in dem Feldgerät 16. Die Verbesserungen der Diagnoseprotokollstruktur erlauben dem Feldgerät 16 die Messung einer Vielzahl von Parametern für eine einzige Prüfkonfiguration. Die Verteilung von Diagnosesteuervorgängen auf die Feldgeräte erlaubt es, Diagnoseprüfungen nach Anforderungen vollständig innerhalb eines Feldgeräts zu steuern, so daß eine Vielzahl
- 30 von Anforderungen an eine Vielzahl von Feldgeräten übermittelt werden kann, die die Diagnoseprüfungen unabhängig voneinander und parallel zueinander steuern. Die Verbesserungen der Diagnoseprotokollstruktur ermöglichen es auch vorteilhafterweise, die Prüfprozeduren außerhalb des Feldgeräts durch Änderungen der Programmierung zu modifizieren. Daher können neue Diagnosekapazitäten hinzugefügt und umfangreiche

Änderungen der Diagnoseoperationen vorgenommen werden, ohne das Feldgerät zu modifizieren. Die Modifikation eines Feldgeräts ist wegen der ungeheuren Kosten des Abschaltens einer Prozeßstraße höchst nachteilig. Die Verbesserungen der Kommunikation ermöglichen es Feldgeräten 16, eine Vielzahl von Parametern für eine einzige Prüfkonfiguration zu messen. Herkömmliche Feldgeräte empfangen eine Anforderung auf Kommunikationsleitungen, die einem bestimmten Feldgerät zugewiesen sind, richten eine Prüfkonfiguration entsprechend den Spezifikationen der Anforderung ein und leiten eine einzige Prüfmessung entsprechend der Anforderung zurück. Eine anschließende Messung unter der gleichen Prüfkonfiguration erfolgt durch Wiederholen des Schritts einschließlich des redundanten Schritts des Einrichtens der Prüfkonfiguration. Die verbesserten Kommunikationen des dargestellten Feldgeräts 16 erlauben vorteilhafterweise die diagnostische Prüfung einer Vielzahl von Geräten parallel miteinander, wodurch der Diagnoseprüfdurchsatz erhöht wird. Ferner liefern die verbesserten digitalen Kommunikationen eine Vielzahl von Datentypen, wogegen dies bei analogen Kommunikationen (von herkömmlichen Feldgeräten) nicht der Fall ist.

Die Erfindung wird zwar unter Bezugnahme auf verschiedene Ausführungsformen beschrieben, es versteht sich jedoch, daß diese Ausführungsformen beispielhaft sind und der Umfang der Erfindung nicht darauf beschränkt ist. Viele Abwandlungen, Modifikationen, Hinzufügungen und Verbesserungen der beschriebenen Ausführungsformen sind möglich. Beispielsweise ist das Feldgerät 16 nur ein Beispiel einer geeigneten Kombination aus Steuerventil und Betätiger. Andere geeignete Ventilarten umfassen Gleitspindel-, Drehstopfen-, Drehkugel-, Rückschlag-, Hubscheibenventile sowie andere bekannte Ventiltypen. Andere geeignete Betätiger umfassen Feder und Membran, Feder und Kolben, doppelwirkende Kolben, hydraulische, elektrohydraulische, elektrische oder andere bekannte Betätigertypen, wobei entweder ein Drehschieber oder ein Gleitspindelventil verwendet wird. Somit ist das Feldgerät 16 nur ein Beispiel für verschiedene Arten von Positionierern oder anderen Einrichtungen zur Steuerung eines Betätigers und Ventils, um die Leistung zu modulieren. Ferner kann das Feldgerät, in dem das Diagnosesystem der vorliegenden Erfindung verwendet wird oder sich befindet, ein von einem Ventil verschiedenes Gerät sein wie beispielsweise eine Pumpensteuereinheit, ein regelbarer Antrieb usw. Ferner ist das Feldgerät 16 nicht auf den Betrieb in Übereinstimmung mit dem Fieldbus-Standard beschränkt, sondern ist außerdem mit anderen

Kommunikationsstandards wie HART, WORLDFIP, LONWORKS, Profibus und dergleichen anwendbar.

Selbstverständlich können Geräte- und Prozeßdiagnoseprüfungen in der Steuereinheit 102 in Form von Funktionsblöcken oder anderer Software gespeichert und so ausgebildet sein, daß diese Diagnoseprüfungen für die Allgemeinheit unter Verwendung jedes Hauptgeräts wie etwa eines Personalcomputers zugänglich sind, so daß diese Prüfungen implementiert werden können, ohne vom Anwender viel Wissen zu verlangen. Bei einer Ausführungsform sind ein oder mehr leicht zugängliche Diagnoseroutinen in dem Gerät (beispielsweise dem Gerät 16) gespeichert und können zur Ausführung gebracht werden, indem ein Hauptgerät einen einzigen Befehl abgibt, der angibt, welche Prüfung ausgeführt werden soll. Alle für die Prüfung notwendigen Daten sind in dem Gerät gespeichert, und alle für die Prüfung gesammelten Ausgangsdaten werden nach beendeter Prüfung an den Hauptrechner geleitet.

Allgemein gesagt können solche "öffentlichen" Geräte- und Prozeßdiagnosen aufgefordert werden, in Wandlerblöcke zu schreiben, um eine Bewegung beispielsweise von Ventilkomponenten zu bewirken, und von den Sensoren erzeugte Daten unter Verwendung eines Trendbildungsblocks in dem Feldbusprotokoll zu lesen und zu speichern. Da die gesamten von dem Hauptrechner geforderten Informationen in der Gerätebeschreibung des Feldgeräts liegen, ist für die Implementierung eines dieser öffentlichen Diagnosevorgänge kein herstellerspezifisches Wissen notwendig. Durch die Anwendung dieser öffentlichen Diagnosevorgänge kann der Betrieb eines Feldgeräts mit dem Betrieb jedes anderen Feldgeräts verglichen werden, das von anderen Herstellern stammt und die gleichen Diagnosemittel enthält.

Beispielhafte Wellenformen, die bei solchen öffentlichen Diagnosen verwendet werden, sind in den Fig. 10A bis 10C gezeigt. Wie Fig. 10A zeigt, kann eine öffentliche Diagnose das Ventil 109 veranlassen, sich schrittweise rampenförmig in Öffnungsrichtung in Einzelschritten von beispielsweise 1 % für eine bestimmte Anzahl von Schritten (z. B. fünf Schritte) zu bewegen und sich dann schrittweise rampenförmig in Schließrichtung in Einzelschritten von 1 % zu bewegen, bis der Ausgangswert oder die Ausgangsposition des Ventils erreicht ist. Bevorzugt erfolgt ein Schritt zu dem neuen Sollwert augenblicklich, d. h. es gibt keine Geschwindigkeitsbeschränkung, und die Verzögerungsdauer bei

jedem neuen Sollwert ist festgelegt und durch die Größe und die Ansprechcharakteristik des Ventil-/Betätigergeräts bestimmt. Alternativ kann, wie Fig. 10B zeigt, eine öffentliche Diagnose ein Ventil über eine Rampe von 5 % in der Öffnungsrichtung bewegen und das Ventil dann unmittelbar über eine Rampe von 5 % in der Schließrichtung bewegen. Die Rate der Rampen ist bevorzugt festgelegt und auf eine Rate eingestellt, die von der Größe der Ventil-/Betätigereinheit bestimmt ist. Am meisten bevorzugt ist die Rampenrate ungefähr die halbe Maximalrate der Geschwindigkeit für das Gerät. Wie Fig. 10C zeigt, kann ferner eine öffentliche Diagnose ein Ventil in der Öffnungsrichtung (oder Schließrichtung) in einem einzigen Schritt von beispielsweise 5 % aus der aktuellen Ventilposition bewegen, wobei die momentane Schrittdauer so vorgegeben ist, daß keine Ratenbeschränkungen wirksam sind. Diese Prüfung endet mit dem Ventil in einer neuen Ventilgrundlinienposition, die um 5 % über (oder unter) dem Anfangspunkt liegt.

Zur Implementierung der in den Fig. 10A bis 10C gezeigten öffentlichen Diagnose unter Anwendung des Fieldbusprotokolls sendet der Hauptrechner 12 einen Ausführungsbe-
fehl an das Gerät, in dem die öffentliche Diagnose gespeichert ist, der in dem Gerät (z. B. dem Gerät 16) eine Trendliste setzt und dann eine geeignete VCR in dem Gerät 16 zur Trendbildung einrichtet. Als nächstes wird das Verbindungsobjekt des Geräts 16 zur Trendbildung eingestellt, und danach läuft die Diagnoseprüfung ab. Dabei kann der Hauptrechner eine Nachricht für den Anwender anzeigen, die bedeutet, daß die Diagnose gerade abläuft, und der Hauptrechner kann den Status oder Fortschritt der Diagnose lesen, um festzustellen, ob ein Fehlerzustand auftritt. Nachdem das Gerät 16 die Diagnose beendet hat, liest und speichert der Hauptrechner die Trenddaten und schaltet die Trendbildung aus. Der Hauptrechner kann dann eine Analyse der abgerufenen Daten durchführen. Wenn von dem Gerät 16 ein Status von beispielsweise 200 % oder mehr empfangen wird, ist im Diagnoseablauf ein Fehler aufgetreten, und der Hauptrechner kann einen solchen Fehler für den Benutzer anzeigen. Nach Analyse der empfangenen Daten kann der Hauptrechner die aus den gespeicherten Trends oder daraus etwa bestimmten Ergebnissen entwickelten Diagnosedaten für den Benutzer anzeigen. Beispielsweise kann der Hauptrechner 12 die tatsächliche Bewegung des Ventils in Abhängigkeit von einer oder sämtlichen der oben beschriebenen öffentlichen Diagnosewellenformen zusammen mit der Eingangswellenform graphisch darstellen, um die Antwort des Geräts 16 auf die Wellenform zu zeigen.

Bei der Durchführung einer öffentlichen oder anderen Diagnose stellt das Gerät 16 zuerst fest, ob ein Diagnosebefehlssignal empfangen wurde, und prüft, wenn das der Fall ist, daß der Wandlerblock des Geräts zufriedenstellend arbeitet. Wenn ja, richtet das Gerät 16 dann aktualisierte Zeiger für Trenddaten ein, um anzuzeigen, welche Daten in dem Trendblock gespeichert werden sollten, und prüft, daß zur Durchführung der Diagnose ein ausreichender Sollwertbereich verfügbar ist. Um beispielsweise eine Prüfung durchzuführen, die eine Bewegung des Ventils in der Öffnungsrichtung um 5 % verlangt, muß sich das Ventil bei oder unterhalb 95 % seiner maximalen Bewegung befinden. Wenn dieser Bereich nicht verfügbar ist, kann das Gerät 16 eine Fehlernachricht zurück an den Hauptrechner 12 senden.

Wenn kein Fehler aufgetreten ist, läßt das Gerät 16 eine ausgewählte Prüfung ablaufen unter Anwendung von Prüfdauern und Neigungsraten, die aus einer Nachschlagetabelle bestimmt sind (auf der Basis der Größe des Ventils/Betätigers), die in dem Gerät 16 gespeichert ist, und führt die gewünschten Messungen wie etwa die Position des Ventils durch. Bei der Prüfung aktualisiert das Ventil den Prüfstatus mit einer Prozent-Komplettierung von 0 bis 100 %, und wenn ein Fehler detektiert wird, schreibt es einen Fehlercode von größer als 200 % in eine Diagnosezustandsvariable, die von dem Hauptrechner 12 gelesen wird. Am Ende der Prüfung und unter der Annahme, daß keine Fehler erreicht worden sind, schreibt die Diagnoseprüfung einen 100 %-Status in die Diagnoseprüfstatusvariable und liefert dann gesammelte Trenddaten an den Hauptrechner unter Anwendung von normalen Stapeloperationen, die eingerichtet sind, um die Trendbildung entsprechend den Standard-Fieldbusprotokollen auszuführen. Bevorzugt wird die Schleifengeschwindigkeit der Diagnoseprüfung relativ hoch vorgegeben im Vergleich mit den Änderungen innerhalb der Eingangswellenform, so daß ausreichend Trenddaten gesammelt werden, um die Antwort des Geräts 16 auf die Diagnosewellenform adäquat zu bilden. In einem Fieldbusprotokoll, in dem die Häufigkeit der Datenstichprobennahme in einem Trendobjekt mit der Ausführungsrate der Funktionsblöcke verknüpft ist,, sollte daher die Schleifenausführungsrate viel höher als die Rate von Änderungen innerhalb der Eingangswellenform sein, um dem Trendobjekt das Sammeln von genügend Daten zu ermöglichen, um die Antwort des Geräts 16 auf die Eingangswellenform nach jeder signifikanten Änderungen in dieser zu beobachten.

Selbstverständlich eignen sich die beispielsweise in Fig. 10 gezeigten öffentlichen Diagnosen am besten zur Durchführung von Gerätediagnosen, da sie vorteilhafterweise genutzt werden können, um ein bestimmtes Gerät zu veranlassen, ein oder mehr Diagnoseschritte oder -operationen zu durchlaufen, während welcher die Ausgänge von Wandlern innerhalb des Geräts gemessen werden, um Charakteristiken dieses Geräts zu bestimmen. Bei Verwendung in einem Fieldbusprotokoll für Gerätediagnosen erfordern diese Prüfungen nicht die Verwendung von irgendwelchen zusätzlichen Funktionsblöcken, sondern können statt dessen von einer einzelnen Gerätesteureinheit (wie etwa der Steuereinheit 102 von Fig. 6) ausgeführt werden, um die Wandlerblöcke getrennt von dem Normalbetrieb zu steuern, der den innerhalb des Geräts wirksamen Funktionsblöcken zugeordnet ist. Falls gewünscht, kann natürlich ein Funktionsblock dazu verwendet werden, die gesammelten Daten auf irgendeine Weise zu analysieren und die analysierten Daten an den Hauptrechner zu übermitteln.

In Fig. 11A ist eine Prozeßsteuerschleife 400 im einzelnen gezeigt, die imstande ist, Geräte- oder Prozeßdiagnosen lokal ausgehend von einem Gerät in einem Fieldbus-Prozeßsteuerungsnetz zu implementieren. Die Schleife 400 weist die Steuerschleife LOOP1 von Fig. 4 auf, in der der AI-Funktionsblock 66 des Geräts 20 kommunikativ über den Bus 34 mit dem PID-Funktionsblock 64 und dem AO-Funktionsblock 63 des Geräts 16 verbunden ist. Die Schleife 400 umfaßt ferner einen Datensammelfunktionsblock 401, der so ausgebildet ist, daß er Daten von dem AO-Funktionsblock 63, dem AI-Funktionsblock 66 und, falls gewünscht, anderen Funktionsblöcken wie etwa AI-Funktionsblöcken 402 und 404 empfängt, die in anderen in das Prozeßsteuerungsnetz eingefügten Feldgeräten angeordnet sein können. Selbstverständlich sind die Funktionsblöcke 66, 402 und 404 so konfiguriert, daß sie mit dem Datensammelfunktionsblock 401 unter Anwendung von synchronen Standardkommunikationen in Verbindung treten können, etwa Veröffentlichter-/Teilnehmer-Kommunikationen, die in dem Fieldbusprotokoll definiert sind.

Im Betrieb unterbricht die Steuereinheit 102 innerhalb des Feldgeräts 16 den Normalbetrieb der Schleife, die aus den Funktionsblöcken 66, 64 und 63 gebildet ist, und liefert eine Diagnosewellenform an den Eingang des AO-Funktionsblocks 63. Dabei wird der Statusmodus des AO-Funktionsblocks 63 beispielsweise zu einem lokalen Steuermodus geändert, der zurück zu dem PID-Funktionsblock 64 kaskadiert und bewirkt, daß

der PID-Funktionsblock 64 seinen Status beispielsweise zu manuell ändert, wodurch wiederum der PID-Funktionsblock 64 daran gehindert wird, ein Ausgangssignal auf der Basis der von ihm empfangenen Eingänge zu erzeugen. Wie oben erwähnt, kann die Diagnosewellenform in der Steuereinheit 102 des Geräts 16 gespeichert sein oder von dem Hauptrechner 12 vor der Implementierung der Diagnoseprüfung geliefert werden. Natürlich bewirken die Wellenform oder andere Anweisungen, daß das dem Gerät 16 zugeordnete Ventil eine Serie von Bewegungen durchläuft, die einer Geräte- oder einer Prozeßdiagnoseprüfung zugeordnet sind.

Während einer Gerätediagnose empfängt der Datensammelfunktionsblock 401 Daten von dem AO-Funktionsblock 63 sowie von anderen Wandlern innerhalb des Geräts 16 wie etwa von Wandlern, die Positionssensoren und/oder irgendwelchen Drucksensoren zugeordnet sind, wie sie in Fig. 6 gezeigt sind. Während einer Prozeßdiagnose empfängt der Datensammelfunktionsblock 401 auch oder alternativ Daten, die auf Prozeßvariablen bezogen sind, die von den AI-Funktionsblöcken 66, 402, 404 sowie irgendwelchen anderen Funktionsblöcken innerhalb des Prozeßsteuerungsnetzes 10 entwickelt werden. Der Datensammelfunktionsblock 401 kann diese Daten gemeinsam mit Daten sammeln, die den Zeitpunkt und die Größe der Diagnosewellenform betreffen, die dem AO-Funktionsblock 63 zugeführt wird, und kann diese Daten zur späteren Abgabe an den Hauptrechner 12 speichern, in dem sie verarbeitet und für einen Benutzer dargestellt werden können.

Nachdem die Geräte- oder Prozeßdiagnoseprüfung beendet ist, werden die von dem Funktionsblock 401 gesammelten Daten dem Hauptrechner 12 zugeführt, und die Steuereinheit 102 übergibt die Steuerung des AO-Funktionsblocks 63 wieder an den PID-Funktionsblock 64 zurück, um die die Blöcke 66, 64 und 63 enthaltende Schleife wieder in den Normalbetrieb zu bringen. Selbstverständlich wird zu diesem Zeitpunkt der Statusmodus der Blöcke 64 und 63 in den Normalbetrieb zurückgebracht.

Es ist zu beachten, daß zwar der Datensammelfunktionsblock 401 nicht unbedingt eine geplante synchrone Ausführungszeit erfordert, die Kommunikationsverbindungen des Funktionsblocks 401 jedoch eingerichtet werden müssen, wenn die Prozeßsteuerschleife 400 konfiguriert wird, und diese Kommunikationsverbindungen existieren somit auch dann, wenn keine Diagnose abläuft. Mit anderen Worten: Wenn der Datensammelfunk-

tionsblock 401 Daten benötigt, die ihm auf dem Bus 34 übermittelt werden, muß er zum Empfang dieser Daten (d. h. der von anderen Geräten veröffentlichten Daten) bei der Initialisierung des Prozeßsteuerungsnetzes 10 konfiguriert werden, um zu vermeiden, daß ein Benutzer das Netz 10 nur zur Durchführung einer Diagnose neu konfigurieren muß. Da jedoch, wie oben gesagt, der Datensammelfunktionsblock 401 im allgemeinen nur Prozeßvariablendaten sammelt, die bereits auf dem Bus 34 veröffentlicht sind, oder Daten sammelt, die intern innerhalb des Geräts erzeugt werden, in dem der Datensammelfunktionsblock 401 vorhanden ist, erweitert die Operation des Datensammelfunktionsblocks 401 im allgemeinen nicht die Menge an synchronen Kommunikationen auf dem Bus 34 bzw. verlangt keine zusätzlichen geplanten Ausführungszeiten für das Prozeßsteuerungsnetz 10. Falls gewünscht, kann jedoch der Datensammelfunktionsblock 401 so konfiguriert werden, daß er Daten empfängt, die gewöhnlich nicht auf dem Bus veröffentlicht werden, was es erforderlich macht, daß der Veröffentlichungsblock so konfiguriert wird, daß diese Daten beim Anfahren des Prozeßsteuerungsnetzes 10 veröffentlicht werden.

Fig. 11B zeigt im einzelnen eine Prozeßsteuerschleife 406, die imstande ist, lokale Diagnosen innerhalb eines Geräts in einem Fieldbus-Prozeßsteuerungsnetz zu implementieren. Ähnlich wie die Schleife 400 von Fig. 11A umfaßt die Schleife 406 die Steuerschleife LOOP1 von Fig. 4, in der der AI-Funktionsblock 66 des Geräts 20 kommunikativ über den Bus 34 mit dem PID-Funktionsblock 64 und dem AO-Funktionsblock 63 des Geräts 16 verbunden ist. Die Schleife 406 umfaßt ferner einen Datensammelfunktionsblock 408 (der sich in dem Gerät 16 befindet), der im wesentlichen gleich wie der Datensammelfunktionsblock 401 wirksam ist mit der Ausnahme, daß er so konfiguriert ist, daß er Daten nur lokal beispielsweise von dem AO-Funktionsblock 63, dem PID-Funktionsblock 64 oder anderen Wandlern oder Sensoren innerhalb des Geräts wie etwa Drucksensoren, Positionssensoren usw. empfängt. Der Datensammelfunktionsblock 408 ist für die Durchführung der Gerätediagnosen besonders nützlich, wenn die angeforderten Daten im allgemeinen lokal innerhalb eines Geräts erzeugt werden. Da der Block 408 Daten nicht von dem Bus 34 empfangen muß, braucht er nicht anfangs beim Anfahren des Prozeßsteuerungsnetzes konfiguriert zu werden.

Selbstverständlich sind die Datensammelfunktionsblöcke 401 und 408 nur zwei Möglichkeiten für das Sammeln von Diagnosedaten innerhalb eines Prozeßsteuerungsge-

räts, und es gibt viele andere Methoden des Sammelns von solchen Daten in verschiedenen Arten von Prozeßsteuerungsnetzen und -geräten. Beispielsweise können die Daten für ein Gerät oder einen Prozeß durch weitere parasitäre Software gesammelt werden, die sich in einem Gerät befindet, das der Definition eines Fieldbus-Funktionsblocks nicht entspricht.

Fig. 12 ist ein Flußdiagramm 500, das Schritte zeigt, die von einer typischen Prozeßdiagnose beispielsweise unter Verwendung der Schleife 400 von Fig. 11 ausgeführt werden. In Schritt 502 wird eine Datensammeleinrichtung dem Datensammelfunktionsblock 401 zugeführt, und andere für eine Diagnoseprüfung notwendige Daten werden von einem Hauptrechner (z. B. dem Hauptrechner 12) an das Gerät 16 übermittelt, in dem die Diagnose ausgeführt werden soll. Danach wird die Diagnose von einem Benutzer initialisiert, so daß die zu prüfende Schleife von dem Rest des Prozeßsteuerungsnetzes getrennt wird. Bei der Initialisierung ändert das Gerät 16 den Status des AO-Funktionsblocks 63 beispielsweise in einen lokalen Steuermodus, was den PID-Funktionsblock 64 (oder andere an der Aufstromseite befindliche Funktionsblöcke) veranlaßt, ihren Modus automatisch entsprechend den Fieldbus-Steuerungsstandards zu ändern. Danach führt ein Block 506 eine Diagnoseanweisung aus, die in dem Speicher des Geräts 16 gespeichert ist, und der Datensammelfunktionsblock 401 sammelt Diagnose- und Prozeßdaten an dem Block 508. Ein Block 510 bestimmt, ob die Prüfung komplett ist, und wenn dies nicht der Fall ist, erfolgt Rücksprung der Steuerung zu dem Block 506, der den nächsten Diagnosebefehl ausführt. Selbstverständlich kann der nächste Befehl eine Wiederholung des vorhergehenden sein, der beispielsweise ein Gerät veranlaßt, sich in einer einzigen Richtung zu bewegen, bis ein Grenzwert erreicht ist.

Die von den Blöcken 506, 508 und 510 ausgeführte Schleife wird wiederholt, bis die Prüfung abgeschlossen oder bis ein Fehler aufgetreten ist oder an dem Block 510 detektiert wird; zu diesem Zeitpunkt beendet ein Block 512 die Diagnoseprüfung, und ein Block 514 stellt den normalen Betriebsstatus beispielsweise des AO-Funktionsblocks 63 wieder her, was den PID-Funktionsblock 64 veranlaßt, seinen Status wieder zu ändern und den AO-Funktionsblock 63 entsprechend dem normalen Steuerungsbetrieb zu steuern.

Die von dem Datensammelfunktionsblock 401 gesammelten Daten werden dem Hauptrechner 12 zugeführt, der die Daten analysiert und/oder zur Anzeige bringt. Falls gewünscht, kann die Analyse in dem Gerät 16 ausgeführt werden, und die Analyseergebnisse können von dem Hauptrechner 12 zu einer Displayeinheit geliefert und angezeigt werden.

Bei Verwendung der Gerätediagnosemethode gemäß der vorliegenden Erfindung sammelt das Gerät 16 sowohl Geräte- als auch Prozeßdaten, ohne daß eine separate Steuerung durch einen Hauptrechner für die Implementierung oder den Empfang der Daten erforderlich ist. Da die Diagnose innerhalb eines Geräts durchgeführt und von der Steuereinheit dieses Geräts anstatt von einem separaten Hauptgerät gesteuert wird, kann der zeitliche Ablauf der Prüfung präzise gesteuert werden, und während einer Diagnoseprüfung gesammelte Daten können ohne weiteres mit der von dem Gerät ausgeführten Operation zeitlich abgestimmt werden, um eine präzise Übereinstimmung zwischen der Prüfwellenform und den Ergebnissen der Prüfung zu liefern. Bei Implementierung zur Durchführung einer Prozeßdiagnose kann der Datensammelfunktionsblock 401 der Schleife 400 (oder einer ähnlichen Schleife, die einen Datensammelfunktionsblock hat, der zum Sammeln von Daten von ein oder mehr Prozeßsteuerschleifen konfiguriert ist) verwendet werden, um das Schleifenverhalten und nicht nur das Geräteverhalten gezielt zu prüfen, und kann dazu verwendet werden anzuzeigen, ob ein Ventil für eine Schleife geeignet ist oder nicht oder ob andere Aspekte bei der Schleife auftreten, die ihr Gesamtverhalten einschränken. Die Prozeß- und/oder Geräteprüfung kann periodisch implementiert werden, ohne daß ein Prozeß abgeschaltet oder in einen Ruhezustand versetzt werden muß. Wenn natürlich der Datensammelfunktionsblock 401 so konfiguriert ist, daß nur Daten gesammelt werden, die auf der Schleife bereits veröffentlicht sind, ist er durch die Steuerungsabwicklung hinsichtlich der Daten, die gesammelt und für eine Diagnose genutzt werden können, begrenzt.

Die Diagnosefunktionen wurden hier zwar als die Durchführung von Diagnosen an einem oder unter Verwendung eines abstromseitigen AO-Funktionsblocks 63 (der ein Ausgangsfunktionsblock ist) und als Eingänge von einem aufstromseitigen PID-Funktionsblock 64 (der ein Steuerfunktionsblock ist) empfangend und Rückführungen an diesen übermittelnd beschrieben, wobei die Blöcke in einer einfachen Steuerschleifenkonfiguration verbunden sind; der Datensammelfunktionsblock 401 oder eine andere

Diagnosefunktionsroutine der vorliegenden Erfindung kann jedoch im Zusammenhang mit anderen Ausgabefunktionen oder Funktionsblöcken und anderen Steuerfunktionen oder Funktionsblöcken beliebig genutzt und in Steuerschleifen implementiert werden, die eine andere als die in Fig. 11 gezeigte Konfiguration haben.

5

Ein Teil der hier beschriebenen Diagnosevorgänge ist zwar in Form eines Fieldbus-"Funktionsblocks" implementiert, es ist jedoch zu beachten, daß die Diagnosen der vorliegenden Erfindung unter Verwendung anderer Arten von Blöcken, Programmen, Hardware, Firmware usw. implementiert werden können, die anderen Arten von Steuersystemen und/oder Kommunikationsprotokollen zugeordnet sind. Tatsächlich verwendet zwar das Fieldbusprotokoll den Ausdruck "Funktionsblock", um einen bestimmten Typ von Einheit zu beschreiben, die imstande ist, eine Prozeßsteuerungsfunktion auszuführen, es ist aber zu beachten, daß der Ausdruck Funktionsblock im vorliegenden Zusammenhang nicht so beschränkt ist und jede Art von Gerät, Programm, Routine oder sonstiger Einheit umfaßt, die imstande ist, eine Prozeßsteuerungsfunktion auf irgendeine Weise an verteilten Stellen innerhalb eines Prozeßsteuerungsnetzes auszuführen. Daher können die hier beschriebenen und beanspruchten Diagnosefunktionsblöcke in anderen Prozeßsteuerungsnetzen implementiert werden oder andere Prozeßsteuerungs-Kommunikationsprotokolle oder Abwicklungspläne verwenden (die es entweder bereits gibt oder die in Zukunft entwickelt werden), die nicht das verwenden, was das Fieldbusprotokoll im strengen Sinn als einen "Funktionsblock" kennzeichnet.

Ferner wurden zwar Prozeß- und Gerätediagnoseoperationen als bei der Durchführung von Diagnosen an (oder unter Verwendung von) Positionierer-/Ventilgeräten verwendet beschrieben, es ist aber zu beachten, daß diese Diagnosen an (oder unter Verwendung von) anderen Gerätetypen ausgeführt werden können, beispielsweise an solchen, die bewegbare Elemente wie etwa Register, Gebläse usw. haben. Ebenso sind zwar die hier beschriebenen Diagnosen bevorzugt in Software implementiert, die in einer Prozeßsteuerungseinrichtung gespeichert ist, sie können aber alternativ oder zusätzlich beliebig in Hardware, Firmware usw. implementiert sein. Bei Implementierung in Software können die Diagnoseoperationen der vorliegenden Erfindung in jedem computerlesbaren Speicher gespeichert sein, etwa auf einer Magnetplatte, einer Laserplatte oder einem anderen Speichermedium, in einem RAM, ROM, EPROM usw. eines Computers u. dgl. Ebenso kann diese Software einem Anwender oder einem Gerät auf jede bekannte

25

30

oder gewünschte Weise übermittelt werden, beispielsweise über einen Nachrichten-
übertragungskanal wie etwa eine Telefonleitung, das Internet usw.

Patentansprüche

1. Feldgerät (16), das eine Diagnoseeinrichtung zur Verwendung in einem Prozeßsteuerungsnetz (10), wie beispielsweise ein Feldbussystem enthält, das eine Vielzahl von Einrichtungen (12 bis 32) aufweist, die durch einen schleifengespeisten Digitalkommunikations-Zweidrahtbus (34) kommunizierend gekoppelt sind, wobei das Feldgerät eine Verbinderschnittstelle (142) aufweist, die mit dem Bus (34) verbunden ist, um Digitalkommunikation über den Bus (34) zu ermöglichen, wobei das Feldgerät ferner folgendes enthält:
- 10 einen Speicher (146, 148, 150), der eine Diagnoseprüfroutine speichert, die eine Serie von Diagnoseprüfbefehlen aufweist;
- eine Steuerung (102, 140), die die genannten Diagnoseprüfbefehle ausführt;
- 15 eine Datensammeleinheit (146, 148, 150, 401, 408), die bei einer Diagnoseprüfung erzeugte Diagnosedaten sammelt;
- gekennzeichnet durch
- 20 eine Kommunikationseinheit (144), die die Daten über den Bus (34) zu einer anderen Einrichtung übermittelt,
- einen Diagnoseprogrammierspracheninterpretierer (144), der in die Steuerung (102) integriert ist, wobei der Interpretierer Befehle wie etwa diejenigen, die die Ausführung von Diagnoseprozeßschritten verlangen, interpretiert, wobei der Interpretierer bevorzugt in einem Programmcode implementiert ist, der in einem ROM (148) gespeichert ist, und in einem Mikroprozessor (140) ausgeführt wird.
- 25
2. Feldgerät nach Anspruch 1, das ferner einen Positionierer aufweist, der mit einem Ventil (56, 109) gekoppelt ist, das ein bewegliches Ventilelement (114) hat, und wobei die Diagnoseprüfbefehle die Bewegung des Ventilelements (114) bezeichnen.
- 30

3. Feldgerät nach Anspruch 2, das ferner einen Positionssensor (55, 116) aufweist, der die Position des beweglichen Ventilelements (114) bei der Diagnoseprüfung erfaßt und der Datensammeleinheit ein Positionssignal liefert, das die Position des Ventilelements (114) bezeichnet.

5 4. Feldgerät nach Anspruch 3, wobei der Positionierer eine Druckluftleitung (118) aufweist, die mit einem Strom-Druck-Meßumformer gekoppelt ist, und ferner einen mit der Druckluftleitung (118) gekoppelten Drucksensor (55, 120, 124, 128) aufweist, der den Druck in der Druckluftleitung (118) erfaßt und der Datensammeleinheit ein Drucksignal liefert, das den Druck in der Druckluftleitung (118) bezeichnet.

10 5. Feldgerät nach Anspruch 4, das ferner ein pneumatisches Relais aufweist, das mit der Druckluftleitung (118) zwischen dem Strom-Druck-Meßumformer (104) und einem Ventil (59, 109) gekoppelt ist, und wobei der Drucksensor (55, 120, 124, 128) mit der Druckluftleitung (118) zwischen dem Strom-Druck-Meßumformer (104) und dem pneumatischen Relais (106) gekoppelt ist.

15 6. Feldgerät nach Anspruch 4, das ferner ein pneumatisches Relais (106) aufweist, das mit der Druckluftleitung (118) zwischen dem Strom-Druck-Meßumformer (104) und einem Ventil (59, 109) gekoppelt ist, und wobei der Drucksensor (55, 120, 124, 128) mit der Druckluftleitung (118) zwischen dem pneumatischen Relais (106) und dem Ventil (59, 109) gekoppelt ist.

20 7. Feldgerät nach Anspruch 4, wobei die Druckluftleitung (118) eine Druckzuführleitung (118) aufweist, die mit einem Eingang des Strom-Druck-Meßumformers (104) gekoppelt ist, und der Drucksensor (55, 120, 124, 128) mit der Druckluftzuführleitung (118) gekoppelt ist, um den Druck zu messen, der dem Eingang des Strom-Druck-Meßumformers (104) zugeführt wird.

25 8. Feldgerät nach Anspruch 1, wobei die Steuerung (14, 102) einen Programmierspracheninterpretierer aufweist, der so ausgebildet ist, daß er eine Programmiersprache interpretiert, und wobei die Diagnoseprüfbefehle in der Programmiersprache gespeichert sind.

9. Feldgerät nach Anspruch 8, wobei die Kommunikationseinheit so ausgebildet ist, daß sie Diagnoseprüfbefehle in der Programmiersprache von einer zweiten von der Vielzahl der Einrichtungen (12 bis 32) über den Bus (34) empfängt und die empfangenen Diagnoseprüfbefehle in dem Speicher speichert.

5

10. Feldgerät nach Anspruch 1, wobei das Feldgerät ein Element aufweist, das in einer Öffnungs- und einer Schließrichtung bewegbar ist, und wobei die Diagnoseprüfbefehle bewirken, daß sich das Element durch eine Stufenrampe in der Öffnungs- und der Schließrichtung bewegt.

10

11. Feldgerät nach Anspruch 10, wobei die Stufenrampe Stufen aufweist, die ungefähr gleich einem Prozent eines Bewegungsbereichs des Elements sind.

15

12. Feldgerät nach Anspruch 1, wobei das Feldgerät ein Element aufweist, das in einer Öffnungs- und einer Schließrichtung bewegbar ist, und wobei die Diagnoseprüfbefehle bewirken, daß sich das Element durch eine lineare Rampe in der Öffnungs- und der Schließrichtung bewegt.

20

13. Feldgerät nach Anspruch 12, wobei die Diagnoseprüfbefehle bewirken, daß sich das Element mit einer Rampengeschwindigkeit bewegt, die ungefähr gleich der halben maximalen Bewegungsgeschwindigkeit des Elements ist.

25

14. Feldgerät nach Anspruch 1, wobei das Feldgerät ein Element aufweist, das bewegbar ist, und wobei die Diagnoseprüfbefehle bewirken, daß sich das Element in einer Stufe bewegt.

15. Feldgerät nach Anspruch 14, wobei die Stufe eine Amplitude von ungefähr fünf Prozent eines Bewegungsbereichs des Elements hat.

30

16. Feldgerät nach Anspruch 1, wobei die Steuerung eine Einrichtungssteuerung ist und wobei die Kommunikationseinheit die Diagnoseprüfbefehle von einem zweiten von der Vielzahl von Einrichtungen über den Bus empfängt und die empfangenen Diagnoseprüfbefehle in dem Speicher speichert.

17. Feldgerät nach Anspruch 16, wobei die Diagnoseprüfbefehle in einer Programmiersprache geschrieben sind und die Gerätesteuerung (14, 102) einen Programmierspracheninterpretierer aufweist, der die Diagnoseprüfbefehle interpretiert, um die Diagnoseprüfung auszuführen.

18. Feldgerät nach Anspruch 17, das ferner einen Positionierer aufweist, der mit einem Ventil gekoppelt ist, das ein bewegliches Ventilelement hat, und wobei die Diagnoseprüfbefehle Bewegungen des Ventilelements bezeichnen.

19. Feldgerät nach Anspruch 18, das ferner einen Positionssensor aufweist, der die Position des Ventilelements bei der Diagnoseprüfung erfaßt und der Datensammeleinheit ein Positionssignal liefert, das die Position des Ventilelements bezeichnet.

20. Feldgerät nach Anspruch 19, wobei der Positionierer eine Druckluftleitung (118) aufweist, die mit einem Strom-Druck-Meßumformer (104) gekoppelt ist, und ferner einen mit der Druckluftleitung gekoppelten Drucksensor (55, 120, 124, 128) aufweist, der den Druck in der Druckluftleitung (118) erfaßt und der Datensammeleinheit ein Drucksignal liefert, das den Druck in der Druckluftleitung (118) bezeichnet.

21. Feldgerät nach Anspruch 16, wobei die Kommunikationseinheit so konfiguriert ist, daß sie über den Bus unter Anwendung eines Fieldbus-Protokolls kommuniziert.

22. Feldgerät nach Anspruch 16, wobei die Diagnoseprüfbefehle eine Prozeßdiagnose implementieren und die Kommunikationseinheit so ausgebildet ist, daß sie über den Bus (34) Daten empfängt, die Prozeßvariable betreffen, die von einem anderen von der Vielzahl der Einrichtungen (12 bis 32) gemessen werden.

23. Feldgerät nach Anspruch 1, wobei der Speicher eine Prozeßdiagnoseprüfroutine speichert, die eine Reihe von Diagnoseprüfbefehlen hat, die von dem Feldgerät zu implementieren sind, wobei die Steuerung eine Einstellungssteuerung ist, die die in dem Speicher gespeicherten Prozeßdiagnoseprüfbefehle ausführt, um eine Prozeßdiagnoseprüfung zu implementieren, wobei die Datensammeleinheit Diagnosedata

ten sammelt, die von dem Feldgerät bei der Prozeßdiagnoseprüfung erzeugt werden, und weitere Prozeßdiagnosedaten von einer zweiten von der Vielzahl der Einrichtungen über den Bus empfängt, und wobei die Kommunikationseinheit die gesammelten Diagnosedaten und die weiteren Prozeßdiagnosedaten über den Bus übermittelt, nachdem
5 die Prozeßdiagnoseprüfung abgeschlossen ist.

24. Feldgerät nach Anspruch 23, wobei die Kommunikationseinheit so konfiguriert ist, daß sie über den Bus unter Anwendung eines Fieldbus-Kommunikationsprotokolls kommuniziert, und wobei die Datensammeleinheit ein Funktionsblock (50) ist, der so konfiguriert ist, daß er die weiteren Prozeßdiagnosedaten von der zweiten von der Vielzahl der Einrichtungen (12 bis 32) über den Bus (34) empfängt.
10

25. Feldgerät nach Anspruch 23, wobei der Funktionsblock (50) so konfiguriert ist, daß er die weiteren Prozeßdiagnosedaten über den Bus (34) unter Anwendung von synchronen Kommunikationen empfängt.
15

26. Feldgerät nach Anspruch 23, wobei die Gerätesteureinheit (102) eine Modushandhabungseinheit ist, die den Modus von Komponenten innerhalb einer Prozeßsteuerschleife steuert, die die Prozeßdiagnoseprüfung ausführt.
20

27. Feldgerät nach Anspruch 1 in einem Prozeßsteuerungsnetz (10), das folgendes aufweist: eine Hauptrechnereinrichtung (12), die Befehle erzeugt und Daten empfängt, eine Vielzahl von Feldgeräten (16 bis 22) und einen Bus (34), der die Hauptrechnereinrichtung (12) und die Vielzahl von Feldgeräten (16 bis 22) kommunikativ miteinander verbindet.
25

28. Feldgerät nach Anspruch 27, wobei die Diagnoseprüfroutine eine Prozeßdiagnoseprüfroutine ist und die Datensammeleinheit Prozeßdiagnosedaten von einem zweiten von der Vielzahl Feldgeräten (12 bis 32) über den Bus (34) empfängt.
30

29. Feldgerät nach Anspruch 27, wobei die Steuerung (14, 102) einen Programmierspracheninterpretierer aufweist, der so ausgebildet ist, daß er eine Program-

miersprache interpretiert, und wobei die Diagnoseprüfbefehle in der Programmiersprache gespeichert sind.

5 30. Feldgerät nach Anspruch 29, wobei die Kommunikationseinheit so ausgebildet ist, daß sie die Diagnoseprüfbefehle in der Programmiersprache von der Hauptrechnereinrichtung (12) über den Bus (34) empfängt und die empfangenen Diagnoseprüfbefehle in dem Speicher speichert, bevor die Diagnoseprüfung implementiert wird.

10 31. Feldgerät nach Anspruch 27, wobei jedes von der Vielzahl von Feldgeräten (12 bis 32) imstande ist, eine Prozeßsteuerungsfunktion auszuführen und auf dem Bus (34) unter Anwendung von planmäßigen periodischen Kommunikationen und nicht-periodischen Kommunikationen zu kommunizieren.

15 32. Feldgerät nach Anspruch 27, wobei das eine von der Vielzahl von Feldgeräten (12 bis 32) ein Element aufweist, das in einer Öffnungs- und einer Schließrichtung bewegbar ist, und wobei die Diagnoseprüfbefehle bewirken, daß sich das Element durch eine Stufenrampe in der Öffnungs- und der Schließrichtung bewegt.

20 33. Feldgerät nach Anspruch 32, wobei die Stufenrampe Stufen aufweist, die ungefähr gleich einem Prozent eines Bewegungsbereichs des Elements sind.

25 34. Feldgerät nach Anspruch 27, wobei das eine von der Vielzahl von Feldgeräten (12 bis 32) ein Element aufweist, das in einer Öffnungs- und einer Schließrichtung bewegbar ist, und wobei die Diagnoseprüfbefehle bewirken, daß sich das Element durch eine lineare Rampe in der Öffnungs- und der Schließrichtung bewegt.

30 35. Feldgerät nach Anspruch 34, wobei die Diagnoseprüfbefehle bewirken, daß sich das Element mit einer Rampengeschwindigkeit bewegt, die ungefähr gleich der halben maximalen Bewegungsgeschwindigkeit des Elements ist.

36. Feldgerät nach Anspruch 27, wobei das eine von der Vielzahl von Feldgeräten (12 bis 32) ein Element aufweist, das bewegbar ist, und wobei die Diagnoseprüfbefehle bewirken, daß sich das Element in einer Stufe bewegt.

37. Feldgerät nach Anspruch 36, wobei die Stufe eine Amplitude von ungefähr fünf Prozent eines Bewegungsbereichs des Elements hat.

5 38. Feldgerät nach Anspruch 1, das ferner zur Anwendung des Fieldbus-Kommunikationsprotokolls ausgebildet ist und wobei die Datensammeleinheit ein Datensammelfunktionsblock (401, 408) ist.

10 39. Feldgerät nach Anspruch 38, wobei die Diagnoseprüfung eine Gerätediagnoseprüfung ist und wobei der Datensammelfunktionsblock Daten von dem Feldgerät sammelt.

15 40. Feldgerät nach Anspruch 38, wobei die Diagnoseprüfung eine Prozeßdiagnoseprüfung ist und wobei der Datensammelfunktionsblock mindestens einen Teil der Diagnosedaten von einem weiteren Feldgerät über Kommunikationen über den Bus sammelt

1 / 11

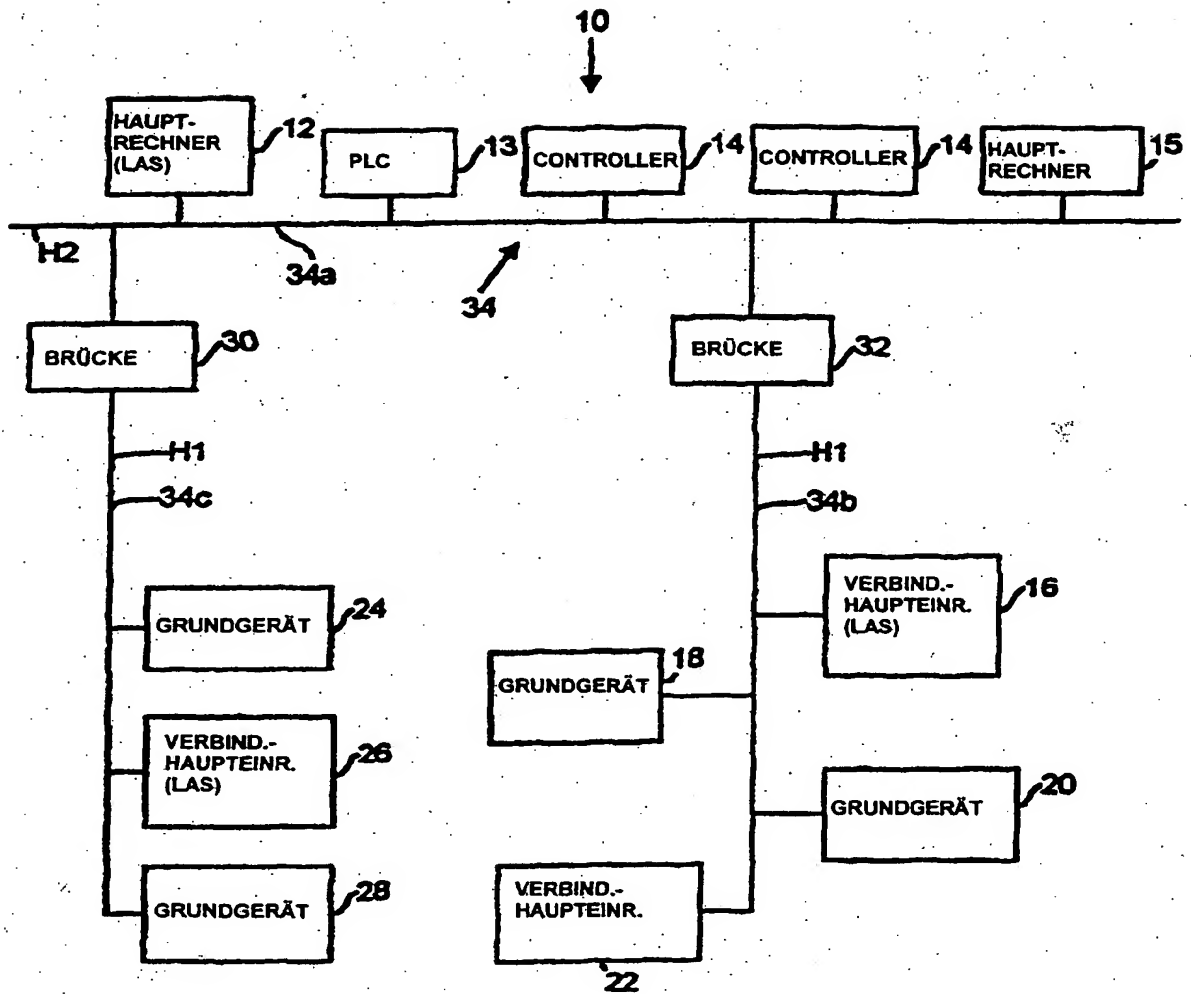


FIG. 1

2/11

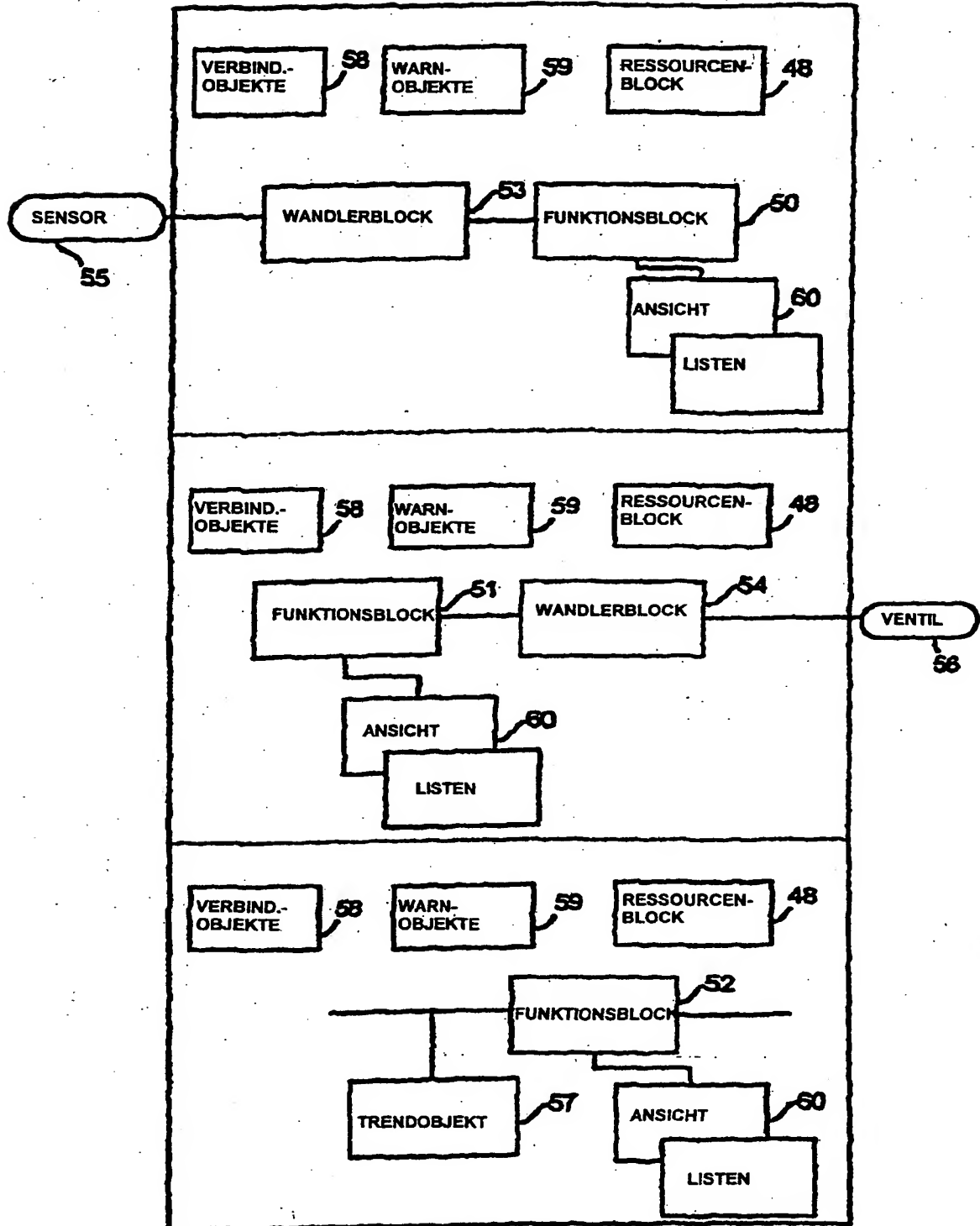


FIG. 2

03.03.03

3 / 11

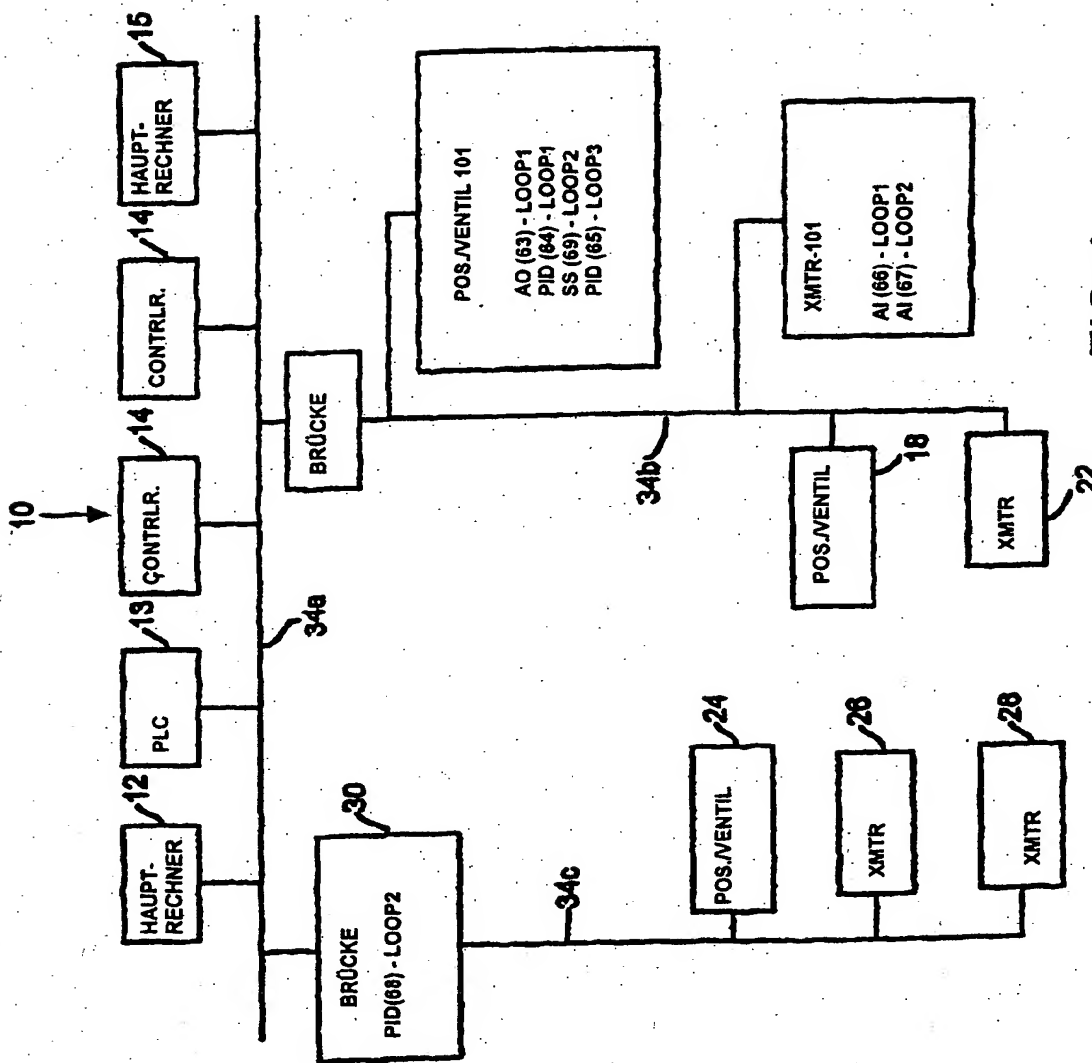


FIG. 3

4/11

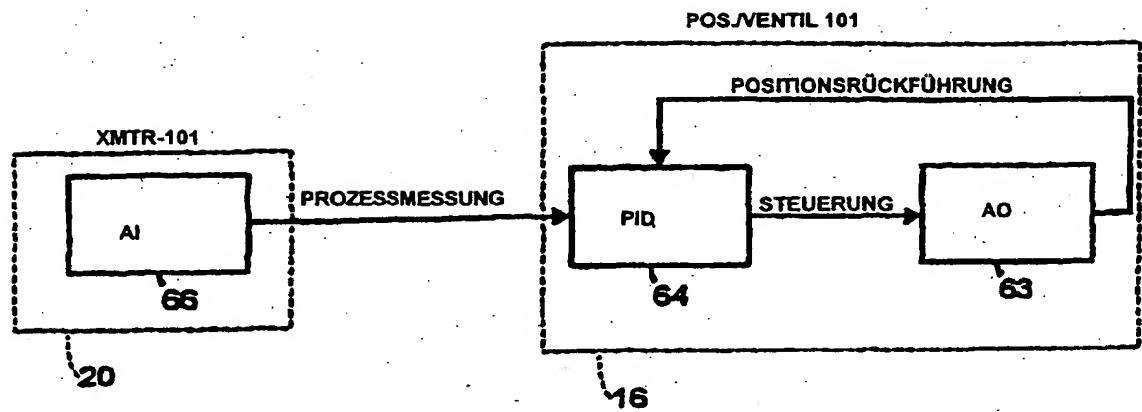


FIG. 4

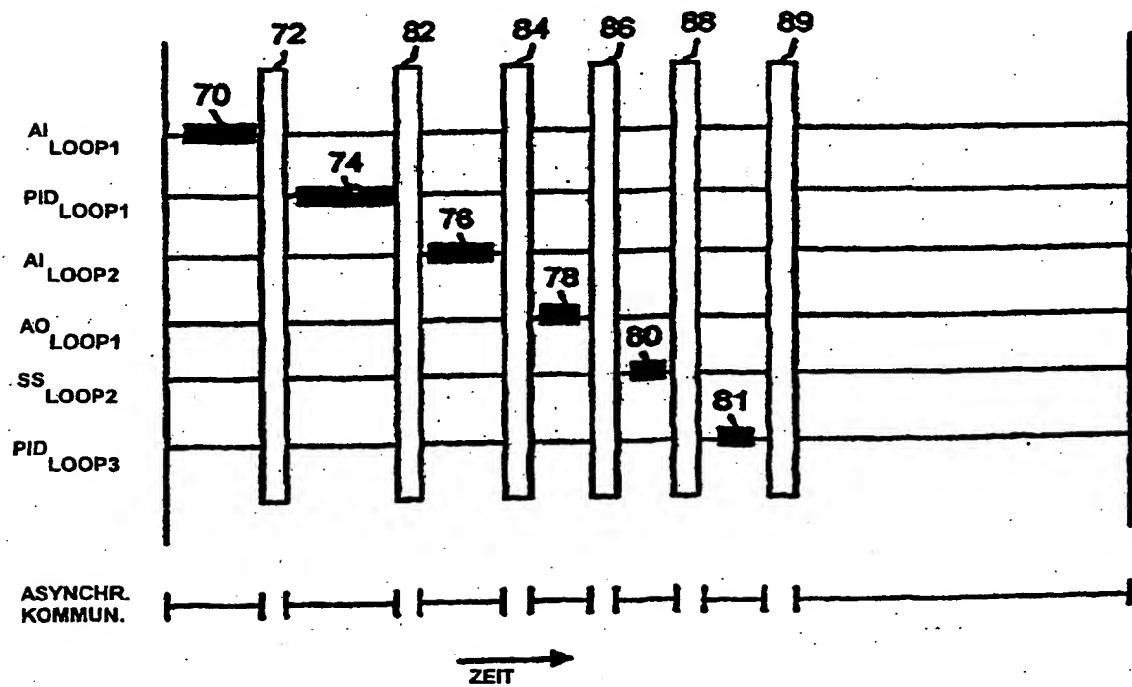


FIG. 5

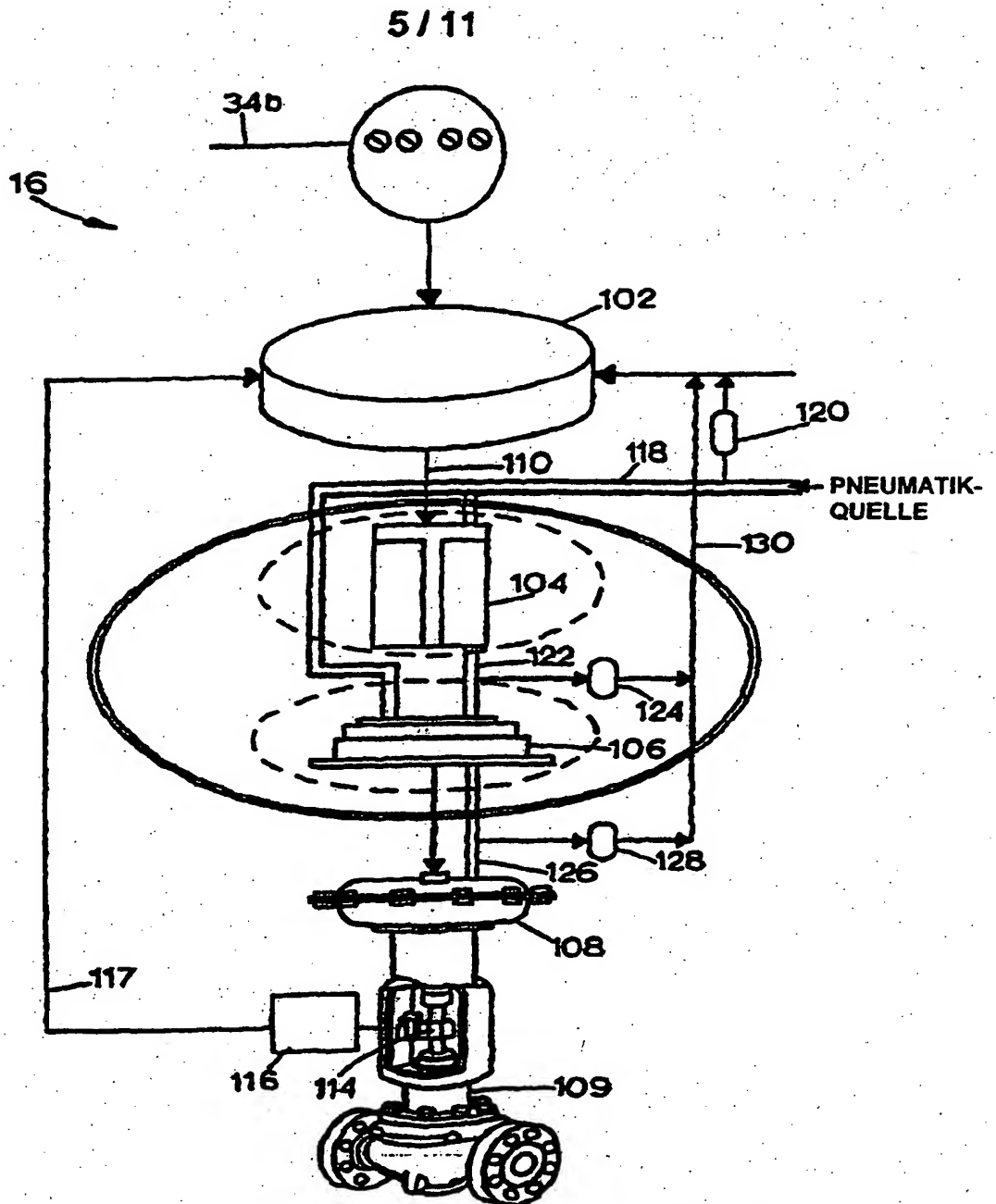


FIG. 6

09.03.03

6/11

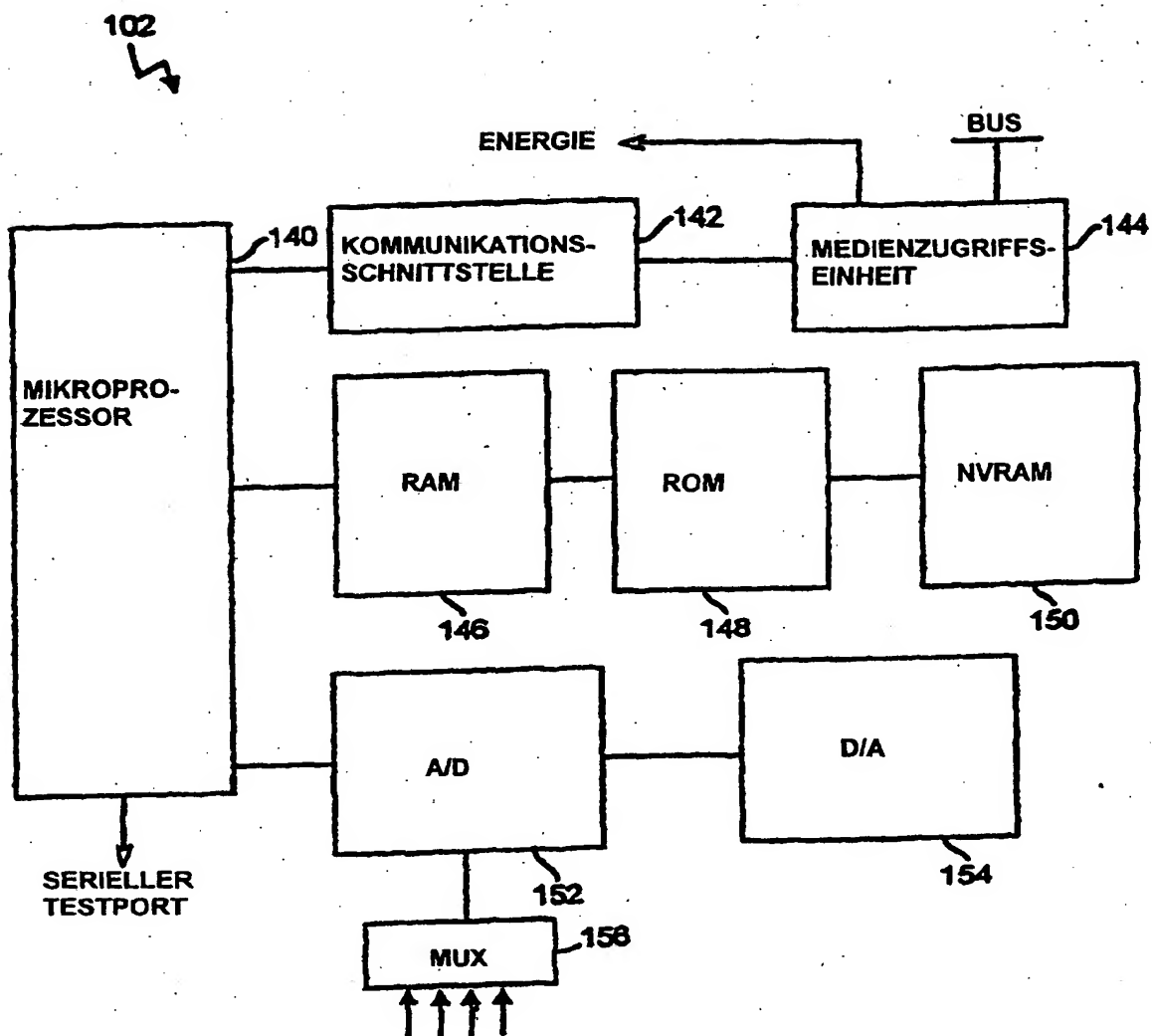


FIG. 7

7/11

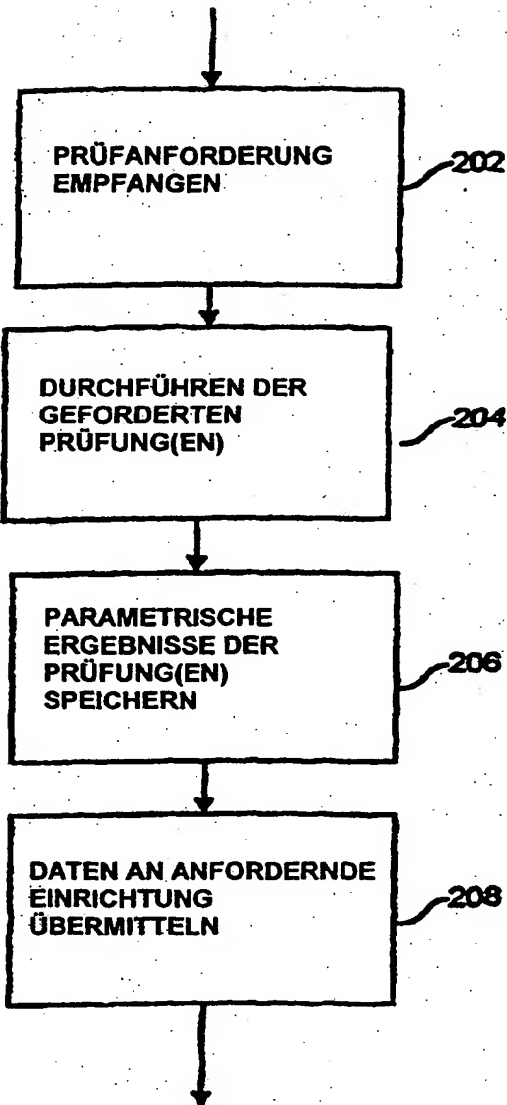
200
↘

FIG. 8

8 / 11

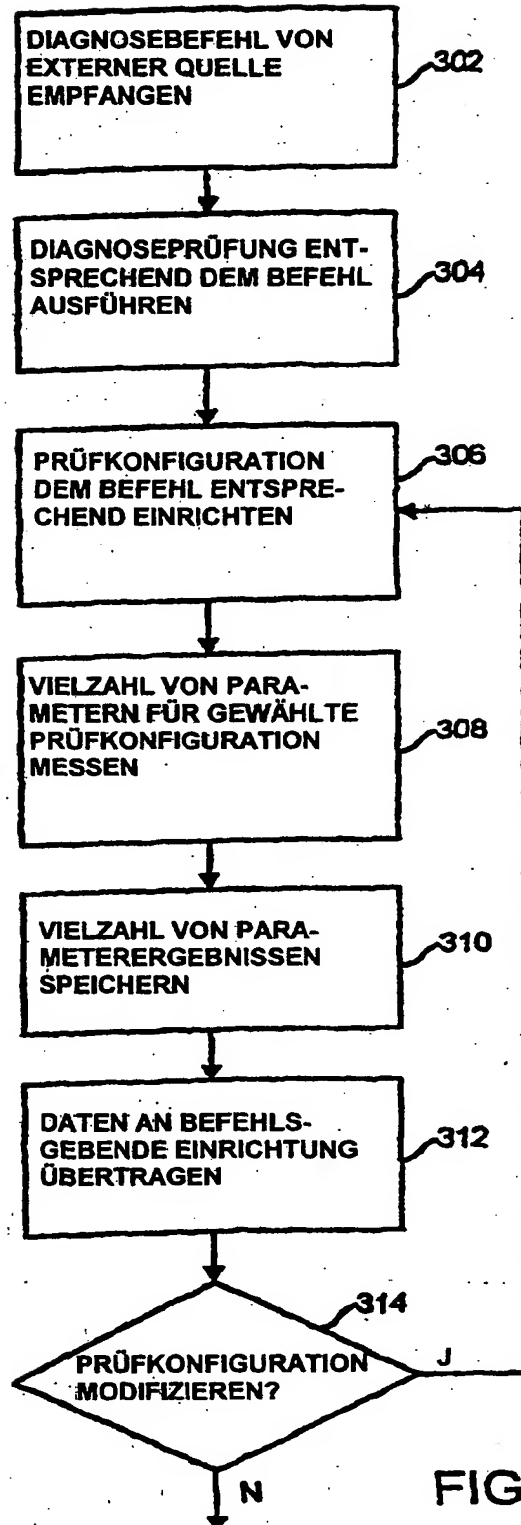
300
↘

FIG. 9

05.03.03

9/11



FIG. 10A

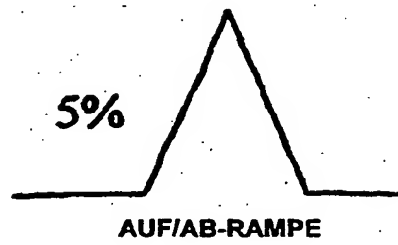


FIG. 10B

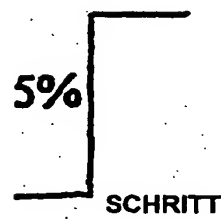


FIG. 10C

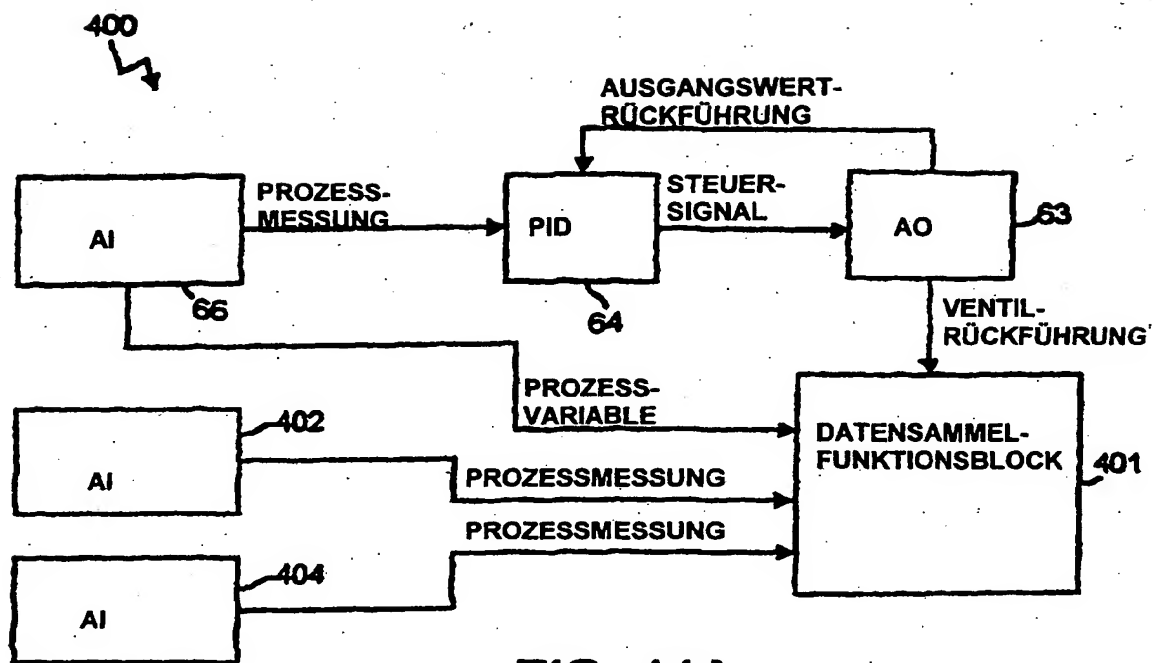


FIG. 11A

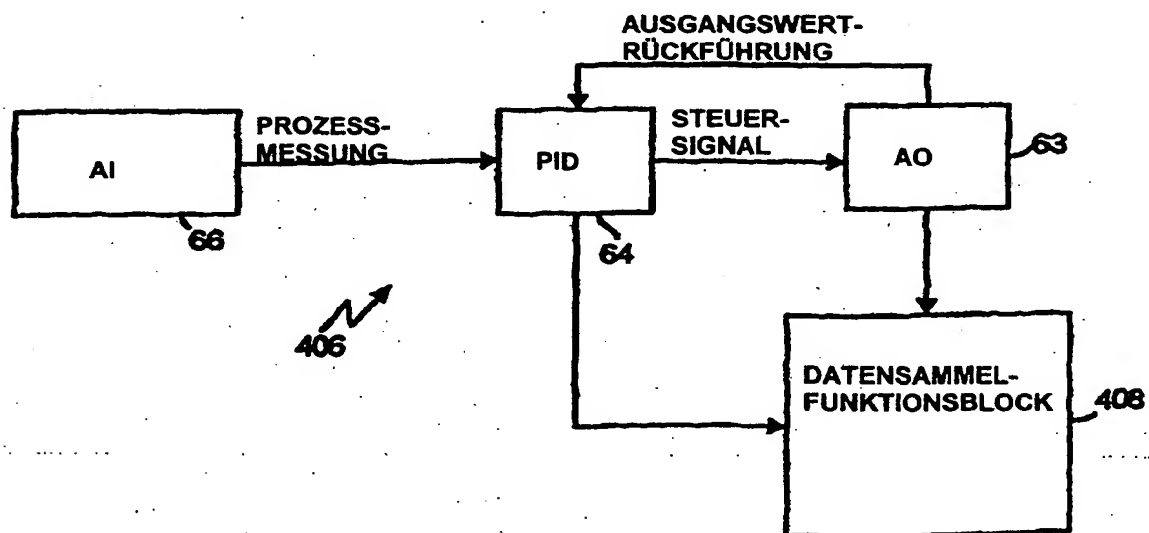


FIG. 11B

11 / 11

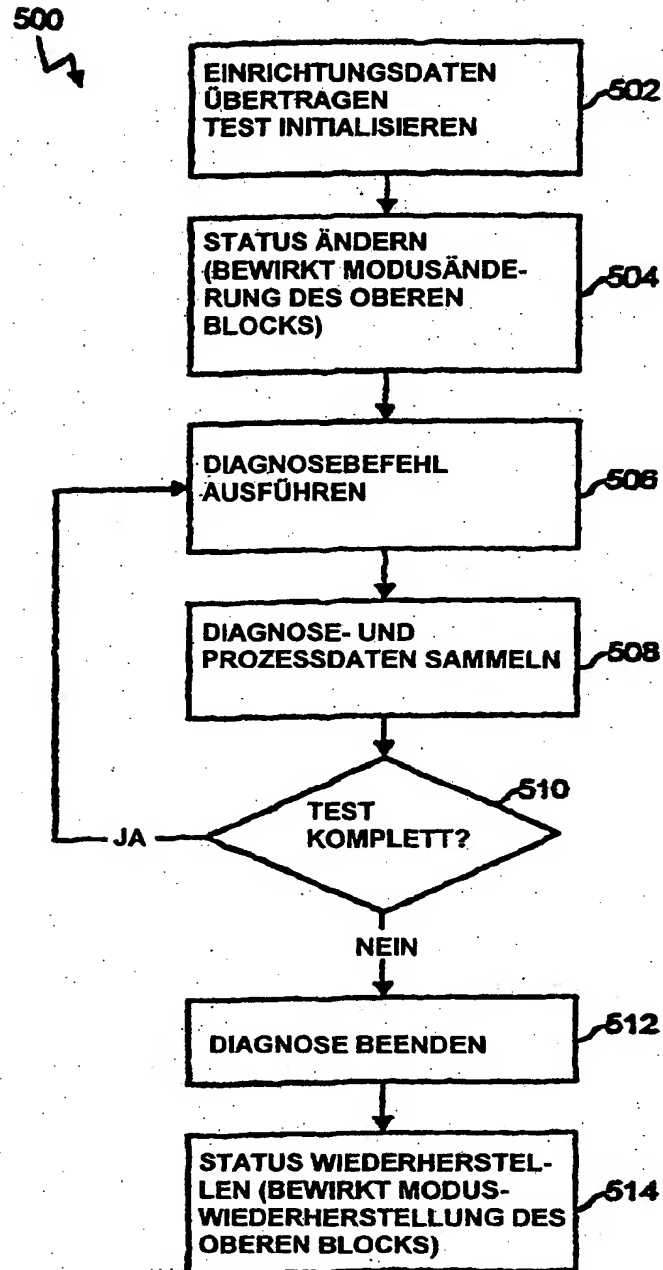


FIG. 12

THIS PAGE BLANK (USPTO)

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.

THIS PAGE BLANK (USPTO)